

**IN THE UNITED STATES DISTRICT COURT
FOR THE DISTRICT OF DELAWARE**

DISTRIBUTED MEDIA SOLUTIONS,
LLC,

Plaintiff,

v.

CURIOSITYSTREAM INC.,

Defendant.

CIVIL ACTION NO. 1:22-cv-01353-GBW

JURY TRIAL DEMANDED

PLAINTIFF’S AMENDED COMPLAINT FOR PATENT INFRINGEMENT

Plaintiff Distributed Media Solutions, LLC (“DMS” or “Plaintiff”) files this Amended Complaint for patent infringement against CuriosityStream Inc. (“CuriosityStream” or “Defendant”) and states as follows:

NATURE OF THE ACTION

1. This is a civil action for patent infringement under the patent laws of the United States of America, 35 U.S.C. § 1 *et seq.*

2. DMS is the owner of all right, title, and interest in U.S. Patent Nos. 6,697,811 (“the ’811 Patent”), 7,133,922 (“the ’922 Patent”), 7,739,714 (“the ’714 Patent”), 8,046,672 (“the ’672 Patent”), and 8,122,004 (“the ’004 Patent”) (collectively, “the Asserted Patents”), which have been attached as Exhibits A-E and incorporated herein by reference.

3. Defendant CuriosityStream has infringed and continues to infringe one or more claims of the Asserted Patents by making, using, offering to sell, and selling within the United States, including in this District, certain products and services, including through its Curiosity product and service. DMS seeks to recover monetary damages, attorneys’ fees, and costs.

THE PARTIES

4. DMS is a Georgia limited liability company with a principal place of business at 4725 Peachtree Corners Circle, Suite 230, Peachtree Corners, GA 30092.

5. On information and belief, Defendant CuriosityStream Inc. is a company organized under the laws of Delaware, with its principal place of business at 8484 Georgia Ave., Suite 700, Silver Spring, Maryland. On information and belief, Defendant CuriosityStream offers its products and/or services, including those accused herein of infringement, to customers and potential customers located in Delaware and in this District.

JURISDICTION AND VENUE

6. The Court has subject matter jurisdiction over this action pursuant to 28 U.S.C. §§ 1331 and 1338(a).

7. This Court has personal jurisdiction over CuriosityStream in accordance with due process because, among other things, CuriosityStream does business in this State. For example, CuriosityStream has engaged, and continues to engage, in continuous, systematic, and substantial activities with this State, including the substantial marketing, use, and sale of products and services with this State and this District. This Court has personal jurisdiction over CuriosityStream because it has committed acts giving rise to DMS's claims for patent infringement within and directed to this District and has derived substantial revenue from its goods and services provided to individuals in this State and this District.

8. Defendant has committed and continues to commit acts of infringement in violation of 35 U.S.C. § 271 within the District of Delaware. Defendant has in the past made, used, marketed, distributed, offered for sale, sold, and/or imported infringing products, and performed infringing methods, in the State of Delaware and in this District. Defendant continues to make,

use, market, distribute, offer for sale, sell, and/or import infringing products, and perform infringing methods, in the State of Delaware and in this District. Accordingly, Defendant has in the past engaged, and continues to engage, in infringing conduct within and directed at or from this District. Additionally, Defendant has purposefully and voluntarily placed its infringing products into the stream of commerce with the expectation that its infringing products will be used in this District. The infringing products have been and continue to be distributed to and used in this District. Defendant's acts have caused, and continue to cause, injury to Plaintiff, including within this District.

9. Venue is proper in this District under the provisions of 28 U.S.C. §§ 1391(b), (c), (d) and/or 1400(b) at least because CuriosityStream is a corporation organized under the laws of Delaware.

U.S. PATENT NO. 6,697,811

10. On September 18, 2018, the United States Patent and Trademark Office duly and legally issued the '811 Patent, entitled "Method and System for Information Management and Distribution" after a full and fair examination.

11. Exhibit A is a true and correct copy of the '811 Patent.

12. The '811 Patent is valid and enforceable under United States patent laws.

13. Plaintiff is the owner of the '811 Patent, having received all right, title and interest in and to the '811 Patent from the previous assignee of record.

14. Plaintiff possesses all rights of recovery under the '811 Patent, including the exclusive right to recover for past infringement.

U.S. PATENT NO. 7,133,922

15. On November 7, 2006, the United States Patent and Trademark Office duly and legally issued the '922 Patent, entitled "Method and Apparatus for Streaming of Data" after a full and fair examination.

16. Exhibit B is a true and correct copy of the '922 Patent.

17. The '922 Patent is valid and enforceable under United States patent laws.

18. Plaintiff is the owner of the '922 Patent, having received all right, title and interest in and to the '922 Patent from the previous assignee of record.

19. Plaintiff possesses all rights of recovery under the '922 Patent, including the exclusive right to recover for past infringement.

U.S. PATENT NO. 7,739,714

20. On June 15, 2010, the United States Patent and Trademark Office duly and legally issued the '714 Patent, entitled "System for Transmitting Digital Data Over a Limited Bandwidth Link in Plural Blocks" after a full and fair examination.

21. Exhibit C is a true and correct copy of the '714 Patent.

22. The '714 Patent is valid and enforceable under United States patent laws.

23. Plaintiff is the owner of the '714 Patent, having received all right, title and interest in and to the '714 Patent from the previous assignee of record.

24. Plaintiff possesses all rights of recovery under the '714 Patent, including the exclusive right to recover for past infringement.

U.S. PATENT NO. 8,046,672

25. On October 25, 2011, the United States Patent and Trademark Office duly and legally issued the '672 Patent, entitled "Method and System for Delivering Technology Agnostic Rich Media Content with an Email, Banner Ad, and Web Page" after a full and fair examination.

26. Exhibit D is a true and correct copy of the '672 Patent.

27. The '672 Patent is valid and enforceable under United States patent laws.

28. Plaintiff is the owner of the '672 Patent, having received all right, title and interest in and to the '672 Patent from the previous assignee of record.

29. Plaintiff possesses all rights of recovery under the '672 Patent, including the exclusive right to recover for past infringement.

U.S. PATENT NO. 8,122,004

30. On February 21, 2012, the United States Patent and Trademark Office duly and legally issued the '004 Patent, entitled "Generating and Providing Rich Media Presentations Optimized for a Device Over a Network" after a full and fair examination.

31. Exhibit E is a true and correct copy of the '004 Patent.

32. The '004 Patent is valid and enforceable under United States patent laws.

33. Plaintiff is the owner of the '004 Patent, having received all right, title and interest in and to the '004 Patent from the previous assignee of record.

34. Plaintiff possesses all rights of recovery under the '004 Patent, including the exclusive right to recover for past infringement.

THE ASSERTED PATENTS

35. The claims of the Asserted Patents are directed to patent-eligible, non-abstract subject matter under 35 U.S.C. § 101. They are not directed to abstract ideas, and the claimed

technology consists of ordered combinations of features and functions that, at the time of the invention, were not, alone or in combination, well-understood, routine, or conventional. Instead, one of skill in the art would recognize that the Asserted Patents are directed to technical and unconventional solutions to shortcomings of the prior art.

36. One of skill in the art would recognize the specifications of the Asserted Patents disclose shortcomings in the prior art and further explain, in detail, the technical and unconventional way the claimed inventions resolve or overcome these shortcomings.

37. For example, the ‘811 Patent explains that “private or special purpose networks may be located globally and may be developed and operated by different organizations.” ‘811 Patent at 1:16-18. As part of the solution to this problem, the ‘811 Patent discloses an invention including a “multi-access manager (MAM)” that is coupled to remote source servers and endpoints that provides a “centralized management but decentralized execution of information distribution.” *Id.* at 3:11-21.

38. The ‘922 Patent explains that a disadvantage with conventional streaming technology is that “current streaming techniques use a point-to-point approach that does not scale well with increasing numbers of users.” ‘922 Patent at 1:46-48. The patent seeks to solve this problem, in part, by providing “an intelligent gateway at the edge of a group of users (such as an Intranet) through which requests for streaming data and the resulting data streams all pass.” *Id.* at 5:6-9.

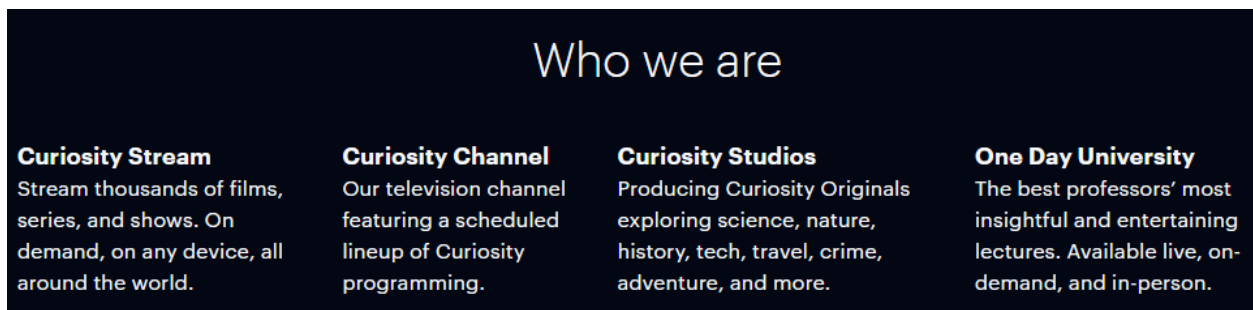
39. The ‘714 Patent notes that “[i]n view of the multitude of bandwidths present in complex client/server systems today and the large amounts of data necessary to produce compelling audio, video and imaging presentations, there is a pressing demand for scalable data representation of multimedia data, so that it can be delivered for on-line interactive playback in

such a form that the transmission rate can accommodate the client bandwidth.” ‘714 Patent at 2:30-37. The patent seeks to solve this problem, in part, by providing “a scalable representation, so that the data can be asynchronously transmitted to clients having different bandwidth connections, played on-line almost immediately after the transmission begins, interactively controlled, and also progressively upgraded as it is replayed.” *Id.* at 18:57-62.

40. The ‘004 and ‘672 Patents explain that one problem with existing systems was that the number of different systems and media files available resulted in users being “constantly bombarded with different requirements” to receive and play those files. ‘004 Patent at 1:20-52. This problem was solved, in part, through the use of an “attribute inquiry system” that determines certain configuration information prior to providing the requested content. *Id.* at 13:28-63.

CURIOSITYSTREAM’S INFRINGING PRODUCTS AND ACTIVITIES

41. CuriosityStream provides Curiosity Stream, an on-demand video-streaming platform (“the Accused Product”) that enables its users to stream video content from multiple sources.



<https://curiositystream.com/about> (last visited 10/03/2022).

42. The Accused Product allows for the streaming of video content on a television, computer, or mobile device.



<https://curiositystream.com/about> (last visited 10/03/2022).

43. In addition, the Accused Product is enabled to be implemented in a browser of a computer. When run on a browser, the Accused Product ascertains certain browser and/or system information to make determinations regarding the content provided to the browser.

44. The Accused Product implements MPEG-DASH to stream video content.

COUNT I: INFRINGEMENT OF U.S. PATENT NO. 6,697,811

45. Plaintiff incorporates by reference and re-alleges the foregoing paragraphs of this Complaint as if fully set forth herein.

46. Defendant has directly infringed at least claim 1 of the '811 Patent in violation of 35 U.S.C. § 271 *et seq.*, by making, using, offering for sale, or selling in the United States, and/or importing into the United States without authority or license the Accused Product. Further, upon information and belief, Defendant directly infringes at least claim 1 of the '811 Patent through hardware and software systems, such as content delivery systems, that it owns, controls, or licenses.

47. Defendant has directly infringed at least claim 1 of the '811 Patent by performing all the limitations of that claim. For example, claim 1 of the '811 Patent recites:

A method for managing and distributing information with a system, comprising:

receiving from a user at a multi-access manager a request for access to the system;

determining whether the request for access to the system comprises an approved access at the multi-access manager;

displaying in response to an approved access a catalogue of at least one source server, wherein each source server couples to a respective information source, and wherein each source server has authorized the user to access data stored at the coupled respective information source;

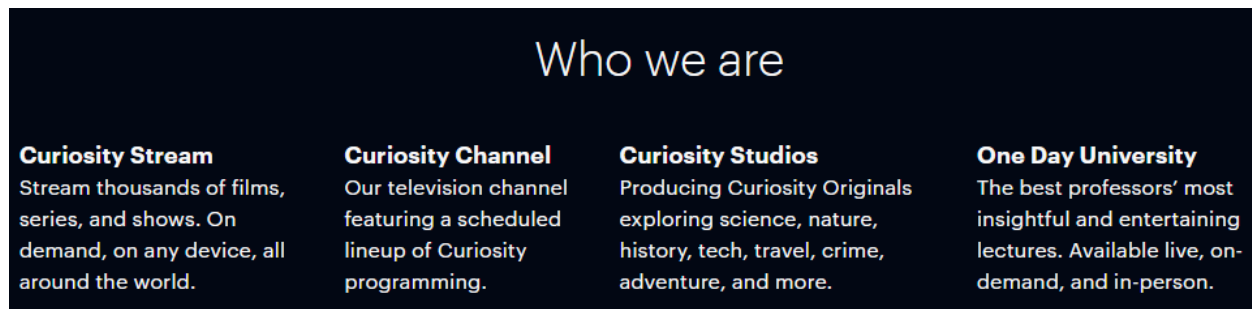
receiving a source server selection from the user, wherein the source server selection comprises a source server selected from the catalogue of at least one source server;

providing access for the user to the selected source server;

receiving a request from the user for data at the selected source server, wherein the request for data requests data accumulated in a first format at the information source coupled to the selected source server; and

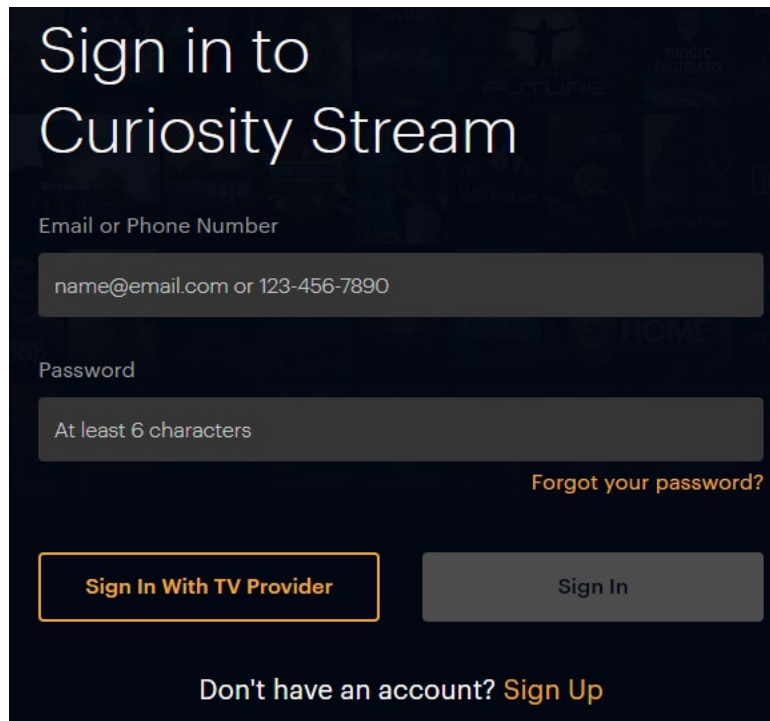
transmitting the data from the selected source server to the user.

48. As in claim 1 of the '811 Patent, the Accused Product practices a method for managing and distributing information (*e.g.*, video content) with a system. Upon information and belief, the Defendant has licensed to use and exercises control over its use of the Akamai Content Delivery Network to implement video streaming services.



<https://curiositystream.com/about> (last visited 10/03/2022). For example, CuriosityStream distributes video content through its CuriosityStream platform.

49. As in claim 1 of the '811 Patent, the Accused Product receives from a user at a multi-access manager (e.g., the login screen of the Curiosity Stream website or application) a request for access to the system.



<https://curiositystream.com/login> (last visited 10/04/2022). For example, as shown, CuriosityStream has a login screen that allows a user to access the CurisotiyStream platform that includes access to multiple video source libraries. Upon information and belief, the CuriosityStream platform, including the CuriosityStream client applications, receives from a user at a log-in screen a request for access to the video distribution system, which includes multiple video libraries.

50. As in claim 1 of the '811 Patent, the Accused Product determines whether the request for access to the system comprises an approved access at the multi-access manager. For example, the Accused Product determines whether a user has access to the system.

Sign in to
Curiosity Stream

Email or Phone Number

name@email.com or 123-456-7890

Password

At least 6 characters

[Forgot your password?](#)

[Sign In With TV Provider](#) [Sign In](#)

Don't have an account? [Sign Up](#)

<https://curiositystream.com/login> (last visited 10/04/2022). As shown, access to the CuriosityStream system requires a valid login that is required for approved access to the video source libraries. Upon information and belief, the CuriosityStream platform, including the CuriosityStream client applications, determines whether the request for access to the CuriosityStream platform comprises an approved access at login.

51. As in claim 1 of the '811 Patent, the Accused Product displays in response to an approved access a catalogue of at least one source server, wherein each source server couples to a respective information source, and wherein each source server has authorized the user to access data stored at the coupled respective information source. For example, the Accused Product includes collections of video content.

Smartest Bundle FAQ



Curiosity Stream Support

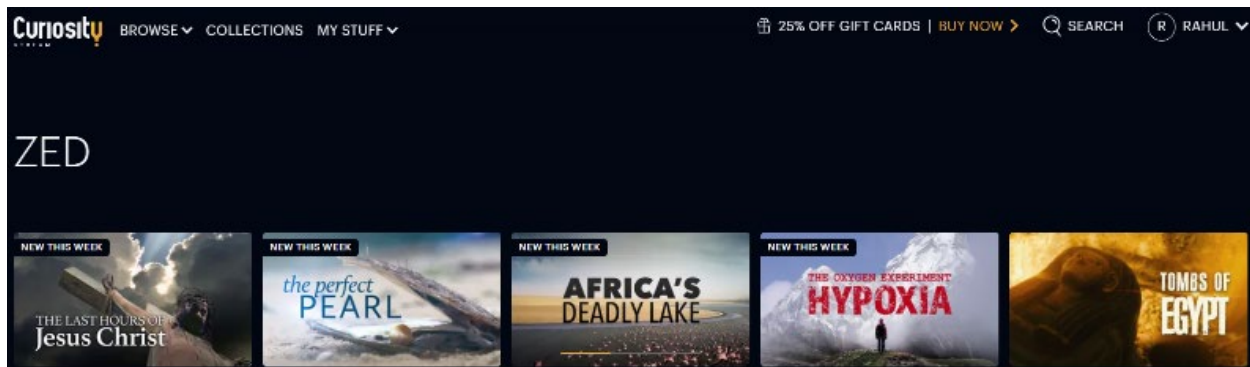
6 days ago · Updated

What's included in the Smartest Bundle (smartbundle.com)?

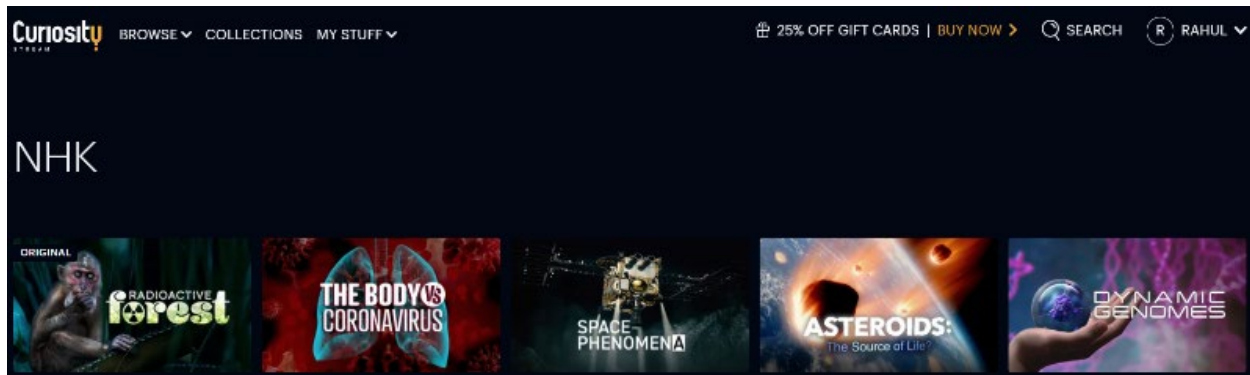
The **Smartest Bundle in Streaming** currently includes subscriptions to Curiosity Stream, Somm TV, Tastemade (US, Canada, United Kingdom, Australia & Germany), Topic (US & Canada), Nebula, and One Day University, with savings of at least 85% compared to subscribing to each service individually.

<https://help.curiositystream.com/hc/en-us/articles/4411394107419-Smartest-Bundle-FAQ> (last visited 10/06/2022). As shown, the access to video content is grouped such that a user can subscribe to bundles of video libraries, which the user can access after logging in. Upon information and belief, the CuriosityStream platform, including the CuriosityStream client applications, includes at least one server that is coupled to and allows access to a video library.

52. As in claim 1 of the '811 Patent, the Accused Product receives a source server selection from the user (*e.g.*, selection of a specific video collection), wherein the source server selection comprises a source server selected from the catalogue of at least one source server.

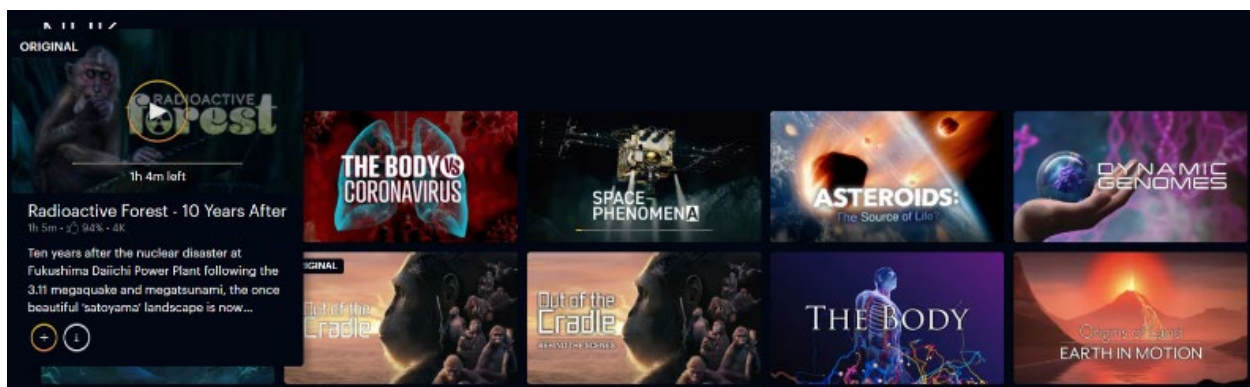


<https://curiositystream.com/browse/producer/ZED> (last visited 10/06/2022).



<https://curiositystream.com/browse/producer/NHK> (last visited 10/06/2022). As shown, the CuriosityStream platform groups content by source library, *e.g.* ZED or NHK. Upon information and belief, the CuriosityStream platform, including the CuriosityStream client applications, allows a user to select a source library from among the available source libraries.

53. As in claim 1 of the '811 Patent, the Accused Product provides access for the user to the selected source server.



<https://curiositystream.com/browse/producer/NHK> (last visited 10/06/2022). Upon information and belief, the CuriosityStream platform, including the CuriosityStream client applications, allows a user to access content from subscribed source libraries.

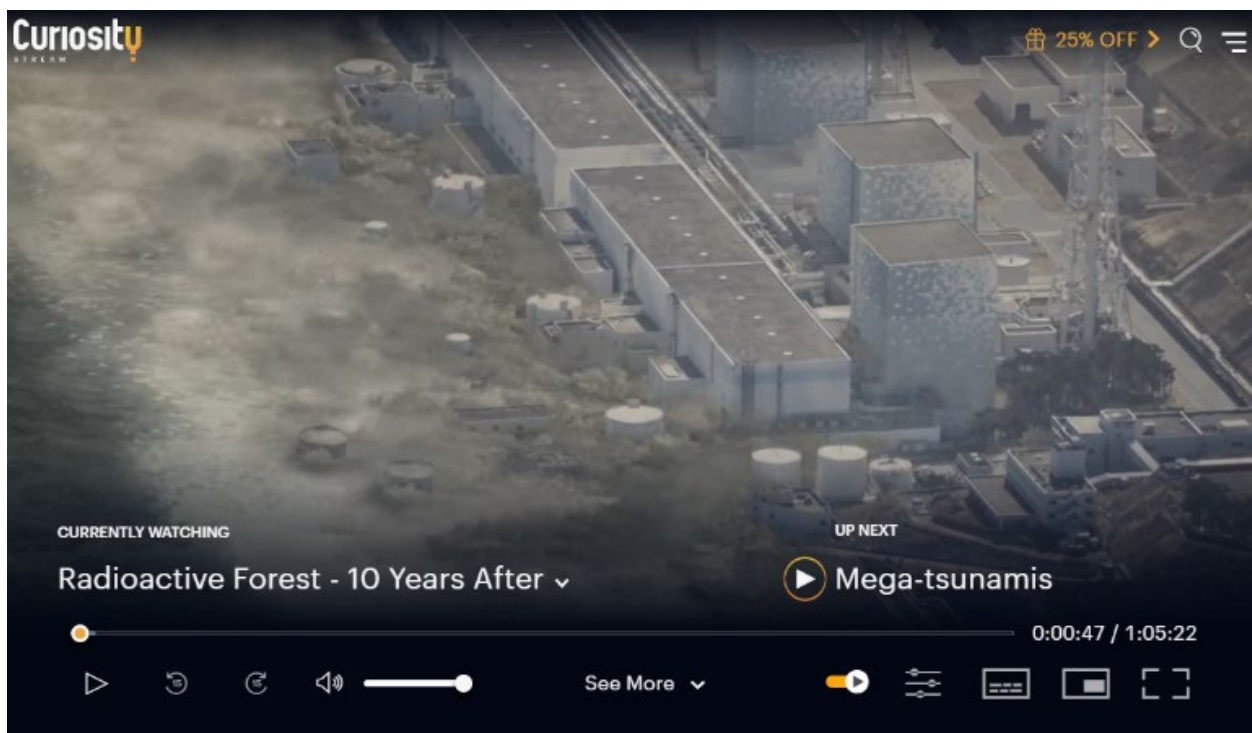
54. As in claim 1 of the '811 Patent, the Accused Product receives a request from the user for data (*e.g.*, video content) at the selected source server, wherein the request for data requests

data accumulated in a first format (e.g., MPEG-DASH) at the information source coupled to the selected source server.

Headers	TextView	SyntaxView	WebForms	HexView	Auth	Cookies	Raw	JSON	XML
<pre>GET https://cdn-s3-akm.curiositystream.com/bitmovin-outputs/NHK_RadioactiveForest_65min_Inq- en_f2997_UHD_6391.mp4/cfd30138cce386399cdb7920e9a51d89.mpd HTTP/1.1 Host: cdn-s3-akm.curiositystream.com Connection: keep-alive sec-ch-ua: "Chromium";v="104", " Not A;Brand";v="99", "Google Chrome";v="104" sec-ch-ua-mobile: ?0 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/104.0.0.0 Safari/537.36 sec-ch-ua-platform: "windows" Accept: */* Origin: https://curiositystream.com</pre>									
Phenomena								View in Notep.	
Transformer	Headers	TextView	SyntaxView	ImageView	HexView	WebView	Auth	Caching	Cookies
<pre>Raw JSON XML HTTP/1.1 200 OK x-amz-id-2: Aawj/14VJb+QWdYwSkMBBLmS9mHRVWtzk39D4AJfbqfpoPmneHgzhWLWYUNAAD2uNFTOI3M/xY= x-amz-request-id: 105KN6MH4CRN3VDF Last-Modified: Fri, 29 Apr 2022 19:09:45 GMT ETag: "ca06f3bbe4392c30423e753b1aa97858" x-amz-version-id: e.n81BG7v3upWBLcuuIRVrhgAJuhSogw Accept-Ranges: bytes Server: AmazonS3 Content-Length: 3351 Cache-Control: max-age=30140019 Expires: Mon, 07 Aug 2023 05:18:55 GMT Date: Tue, 23 Aug 2022 09:05:16 GMT Connection: keep-alive Vary: Accept-Encoding Akamai-Mon-Iucid-De1: 949116 Content-Type: application/dash+xml Access-Control-Max-Age: 86400 Access-Control-Allow-Credentials: true Access-Control-Expose-Headers: Server,range,hdntl,hdnts,Akamai-Mon-Iucid-Ing,Akamai-Mon- Iucid-De1,Akamai-Request-BC Access-Control-Allow-Headers: origin,range,hdntl,hdnts Access-Control-Allow-Methods: GET,POST,OPTIONS Access-Control-Allow-Origin: *</pre> <pre><?xml version="1.0" encoding="UTF-8" standalone="yes"?> <MPD id="b1d485aa-8385-462f-b9c1-93b5fefd131a" type="static" xmlns="urn:mpeg:dash:schema:mpd:2011" profiles="urn:mpeg:dash:profile:isoff- live:2011,urn:com:dashif:dash264" xmlns:ns2="http://www.w3.org/1999/xlink" minBufferTime="POYOMODTOHOM2.000S" mediaPresentationDuration="POYOMODT1H5M22.852S"><Period id="d6d78c76-9e36-41c3-b14d-f0084d6b7305"><AdaptationSet segmentAlignment="true" mimeType="video/mp4"><Representation id="5fa49ce2-5fad-49a0-bf9f-fe611c7de9a1" width="640" codecs="avc1.64001E" height="360" bandwidth="145000" frameRate="30000/1001"><SegmentTemplate media="video/h264/360/145000/base/seg_\$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/360/145000/base/init.mp4"/></Representation> <Representation id="a56312de-ea9c-4075-9a67-9baf72d0f053" width="640" codecs="avc1.64001E" height="360" bandwidth="275500" frameRate="30000/1001"><SegmentTemplate media="video/h264/360/275500/base/seg_\$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/360/275500/base/init.mp4"/></Representation> <Representation id="02baf0c5-90b0-43fa-994d-0848fd709f52" width="640" codecs="avc1.64001E" height="360" bandwidth="456963" frameRate="30000/1001"><SegmentTemplate media="video/h264/360/456963/base/seg_\$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/360/456963/base/init.mp4"/></Representation> <Representation id="15029b83-b46e-4935-81ed-4b15888a10ae" width="960" codecs="avc1.64001E" height="540" bandwidth="735702" frameRate="30000/1001"><SegmentTemplate media="video/h264/540/735702/base/seg_\$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/540/735702/base/init.mp4"/></Representation> <Representation id="f7d81b6c-1848-441c-a5a9-80e787b79e4f" width="1280" codecs="avc1.64001F" height="720" bandwidth="1397455" frameRate="30000/1001"><SegmentTemplate</pre>									

As shown, the CuriosityStream platform, upon receiving a request for video content, streams the requested content to a user in an MPEG-DASH format. Upon information and belief, the CuriosityStream platform, through at least its use of the Akamai CDN, which upon information and belief is a licensed and controlled component of the CuriosityStream platform, along with the CuriosityStream client applications receives a request from the user for video content from a selected video content library, wherein the request is for video content in MPEG-DASH format in the video content library.

55. As in claim 1 of the '811 Patent, the Accused Product transmits the data from the selected source server to the user.



<https://curiositystream.com/video/6391> (last visited 10/06/2022). Upon information and belief, the CuriosityStream platform, through at least its use of the Akamai CDN, which upon information and belief is a licensed and controlled component of the CuriosityStream platform, along with the CuriosityStream client applications streams the selected content from the source library to the user.

56. Defendant makes, uses, sells, and/or offers to sell the Accused Product which practices at least claim 1 of the '811 Patent.

57. In violation of 35 U.S.C. § 271, Defendant has been directly infringing the '811 Patent, including through its own use, testing, and sale of the Accused Products.

58. Defendant has had knowledge of infringement of the '811 Patent at least as of the service of the present Complaint.

59. Defendant has directly infringed at least one claim of the '811 Patent by making, using, offering for sale, and selling the Accused Product without authority in the United States. As a direct and proximate result of Defendant's direct infringement of the '811 Patent, DMS has been damaged.

60. By engaging in the conduct described herein, Defendant has injured DMS and is thus liable for infringement of the '811 Patent, pursuant to 35 U.S.C. § 271.

61. Defendant has committed these acts of infringement without license or authorization.

62. As a result of Defendant's infringement of the '811 Patent, Plaintiff has suffered monetary damages and is entitled to a monetary judgment in an amount adequate to compensate for Defendant's past infringement, together with interests and costs.

63. Plaintiff reserves the right to modify its infringement theories as discovery progresses in this case; it shall not be estopped for infringement contention or claim construction purposes by the claim chart that it provides with this Complaint.

COUNT II: INFRINGEMENT OF U.S. PATENT NO. 7,133,922

64. Plaintiff incorporates by reference and re-alleges the foregoing paragraphs of this Complaint as if fully set forth herein.

65. Defendant has directly infringed at least claim 18 of the '922 Patent in violation of 35 U.S.C. § 271 *et seq.*, by making, using, offering for sale, or selling in the United States, and/or importing into the United States without authority or license the Accused Product. Further, upon information and belief, Defendant directly infringes at least claim 18 of the '922 Patent through hardware and software systems, such as content delivery systems, that it owns, controls, or licenses.

66. Defendant has directly infringed at least claim 18 of the '922 Patent by performing all the limitations of that claim. For example, claim 18 of the '922 Patent recites:

A method for providing streaming data from a server to multiple clients, comprising

locating a plurality of gateways between said server and said clients, each said client being associated with one said gateway,

sourcing a data stream from a server or another gateway in the event that a request for a stream is a first request to a said gateway for a said stream,

supplying a data stream from the said gateway to a second or subsequent client requesting a data stream,

deciding whether a neighbouring gateway exists from which a first requested data stream may be obtained,

selecting between two or more gateways that are possible sources of a requested data stream by interrogating said possible sources about the loading of the possible source gateways, quality of the data stream and the communication latency between the gateway and the possible source gateways.

67. As in claim 18 of the '922 Patent, the Accused Product practices a method for providing streaming data (*e.g.*, video content) from a server to multiple clients. For example, the Accused Product uses an Akamai Content Delivery Network. Upon information and belief, the Defendant has licensed to use and exercises control over its use of the Akamai Content Delivery Network to implement video streaming services.

Who we are

Curiosity Stream Stream thousands of films, series, and shows. On demand, on any device, all around the world.	Curiosity Channel Our television channel featuring a scheduled lineup of Curiosity programming.	Curiosity Studios Producing Curiosity Originals exploring science, nature, history, tech, travel, crime, adventure, and more.	One Day University The best professors' most insightful and entertaining lectures. Available live, on-demand, and in-person.
--	---	---	--

<https://curiositystream.com/about> (last visited 10/03/2022). For example, upon information and belief, the CuriosityStream platform, through at least its use of the Akamai CDN, which upon information and belief is a licensed and controlled component of the CuriosityStream platform, along with the CuriosityStream client applications streams video content to multiple users.

68. As in claim 18 of the '922 Patent, the Accused Product locates a plurality of gateways between said server and said clients, each said client being associated with one said gateway.

#	Result	Protocol	Host	URL
153	200	HTTPS	cdn-s3-akm.curiositystre...	/bitmovin-outputs/ZED_Trajecto...
154	200	HTTPS	cdn-s3-akm.curiositystre...	/bitmovin-outputs/ZED_Trajecto...
155	200	HTTPS	cdn-s3-akm.curiositystre...	/bitmovin-outputs/ZED_Trajecto...
156	204	HTTPS	m.clarity.ms	/collect

The screenshot displays the 'Headers' tab of a web browser's developer tools. The top section shows the 'Request' headers for a GET request to a video file on a CuriosityStream CDN. The bottom section shows the 'Response' headers, indicating the file is served from Amazon S3 via Akamai, with various security and caching headers.

Request Headers:

```

GET https://cdn-s3-akm.curiositystream.com/bitmovin-outputs/ZED_Trajectory_101_ENG_f25_HD_4732.mp4/video/h264/540/492867/base/seg_2.m4s HTTP/1.1
Host: cdn-s3-akm.curiositystream.com
Connection: keep-alive
sec-ch-ua: "Chromium";v="104", " Not A;Brand";v="99", "Google Chrome";v="104"
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/104.0.0.0 Safari/537.36
sec-ch-ua-platform: "Windows"
Accept: */*
Origin: https://curiositystream.com
Sec-Fetch-Site: same-site
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: https://curiositystream.com/

```

Response Headers:

```

HTTP/1.1 200 OK
x-amz-id-2: oUxwvH1b9dCewEQ36wgbAQtxmebDw+QHEyMrGTYKwJ6fA3u6e8UCb3SpzB0aLodaKWuL+15nh5s=
x-amz-request-id: OVD5YBK2PX8JAORH
Last-Modified: Tue, 03 Aug 2021 14:18:41 GMT
ETag: "0d261d099ee5c2ccbbc57039ac82f2d4"
x-amz-version-id: k2xGB8VBu04HwKtbws2k.uIQVDPNxnYs
Accept-Ranges: bytes
Server: AmazonS3
Content-Length: 579529
Unused62: 8096267
Cache-Control: max-age=30268332
Expires: Mon, 07 Aug 2023 15:32:11 GMT
Date: Mon, 22 Aug 2022 07:39:59 GMT
Connection: keep-alive
Akamai-Mon-Iucid-Del: 949116
Content-Type: video/mp4
Access-Control-Max-Age: 86400
Access-Control-Allow-Credentials: true
Access-Control-Expose-Headers: Server,range,hdntl,hdnts,Akamai-Mon-Iucid-Ing,Akamai-Mon-Iucid-Del,Akamai-Request-BC
Access-Control-Allow-Headers: origin,range,hdntl,hdnts
Access-Control-Allow-Methods: GET,POST,OPTIONS

```

Additionally, for example, upon information and belief, the CuriosityStream platform, through at least its use of the Akamai CDN, which upon information and belief is a licensed and controlled component of the CuriosityStream platform, associates a client to an edge server.

Connecting to an Akamai Edge Server

Fundamentally, the Akamai Intelligent Platform improves user experience by serving your content from an Akamai edge server that is physically close to the user, no matter where in the world that user is. This is why there are so many servers in the Akamai Intelligent Platform.

In order for a user to retrieve your content from an Akamai edge server close to that user, when the user makes a request to www.example.com, somehow that request needs to arrive at the Akamai edge server instead of at www.example.com. This re-routing of requests is done via DNS, the Domain Name System. Customers often publish their content at an Akamai hostname, for example

www.example.com.edgesuite.net. When the browser looks up this hostname, the user's DNS server will contact Akamai's DNS servers. Akamai DNS servers are typically deployed in layers: the first layer is named a Top Level Name Server (TLNS), which talk to the lower layers which are composed of Low Level Name Servers (LLNS).

Our TLNS servers will try to locate an Akamai server deployment near the user. The user's DNS server is used as a proxy for the actual location of the user. Once a server deployment has been chosen, the TLNS server will provide a pointer to the LLNS server within that deployment.

The user's DNS server will then contact the LLNS server in order to learn the IP address of the edge server in that deployment which can serve the customer's content.

After learning the IP address of an edge server, the user will open a connection to that server and request the content.

https://developer.akamai.com/legacy/learn/Overview/Client_Edge_Servers_Origin.html (last

visited 9/20/2022). Upon information and belief, the CuriosityStream platform, through at least its use of the Akamai CDN, which upon information and belief is a licensed and controlled component of the CuriosityStream platform, includes multiple edge servers that can be associated with a user based on their location.

69. As in claim 18 of the '922 Patent, the Accused Product sources a data stream from a server or another gateway in the event that a request for a stream is a first request to a said gateway for a said stream.

Locating an object

The edge server will check its local cache as well as the caches of other machines in the server deployment to see if the requested object has been seen before. If the object is found, the edge server will verify that the object is not stale and will serve it to the user.

If the object is found in the cache but it is stale, the edge server will contact another Akamai deployment or the origin to see if a newer version has been uploaded.

Origin retrieval and tiered distribution

When an edge server gets a request for an object that it hasn't yet seen, it will download it from either another Akamai deployment or the origin. The customer's metadata determines whether the edge contacts the origin directly, or if it applies some sort of tiered distribution hierarchy.

Tiered distribution is used to provide greater origin offload by allowing many Akamai edge deployments to go forward to a smaller set of deployments which in turn go forward to the origin. In the case of Akamai's Site Shield product, the Customer's IT department can program the IP addresses of these top-tier machines into their firewall and block access to their network from all other Internet hosts.

At this point, caching rules are applied to the object and the requested bytes are delivered to the user.

https://developer.akamai.com/legacy/learn/Overview/Client_Edge_Servers-Origin.html (last visited 9/20/2022). As shown, upon information and belief, the CuriosityStream platform, through at least its use of the Akamai CDN, which upon information and belief is a licensed and controlled component of the CuriosityStream platform, sources video content from another edge server or the origin in the event that the request for the video content is the first request at the edge server.

70. As in claim 18 of the '922 Patent, the Accused Product supplies a data stream from the said gateway to a second or subsequent client requesting a data stream.

Locating an object

The edge server will check its local cache as well as the caches of other machines in the server deployment to see if the requested object has been seen before. If the object is found, the edge server will verify that the object is not stale and will serve it to the user.

If the object is found in the cache but it is stale, the edge server will contact another Akamai deployment or the origin to see if a newer version has been uploaded.

Origin retrieval and tiered distribution

When an edge server gets a request for an object that it hasn't yet seen, it will download it from either another Akamai deployment or the origin. The customer's metadata determines whether the edge contacts the origin directly, or if it applies some sort of tiered distribution hierarchy.

Tiered distribution is used to provide greater origin offload by allowing many Akamai edge deployments to go forward to a smaller set of deployments which in turn go forward to the origin. In the case of Akamai's Site Shield product, the Customer's IT department can program the IP addresses of these top-tier machines into their firewall and block access to their network from all other Internet hosts.

At this point, caching rules are applied to the object and the requested bytes are delivered to the user.

https://developer.akamai.com/legacy/learn/Overview/Client_Edge_Servers_Origin.html (last visited 9/20/2022). As shown, upon information and belief, the CuriosityStream platform, through at least its use of the Akamai CDN, which upon information and belief is a licensed and controlled component of the CuriosityStream platform, along with the CuriosityStream client applications provides video content from an edge server to a second or subsequent client requesting video content.

71. As in claim 18 of the '922 Patent, the Accused Product decides whether a neighboring gateway exists from which a first requested data stream may be obtained.

Locating an object

The edge server will check its local cache as well as the caches of other machines in the server deployment to see if the requested object has been seen before. If the object is found, the edge server will verify that the object is not stale and will serve it to the user.

If the object is found in the cache but it is stale, the edge server will contact another Akamai deployment or the origin to see if a newer version has been uploaded.

Origin retrieval and tiered distribution

When an edge server gets a request for an object that it hasn't yet seen, it will download it from either another Akamai deployment or the origin. The customer's metadata determines whether the edge contacts the origin directly, or if it applies some sort of tiered distribution hierarchy.

Tiered distribution is used to provide greater origin offload by allowing many Akamai edge deployments to go forward to a smaller set of deployments which in turn go forward to the origin. In the case of Akamai's Site Shield product, the Customer's IT department can program the IP addresses of these top-tier machines into their firewall and block access to their network from all other Internet hosts.

At this point, caching rules are applied to the object and the requested bytes are delivered to the user.

https://developer.akamai.com/legacy/learn/Overview/Client_Edge_Servers_Origin.html (last visited 9/20/2022). As shown, upon information and belief, the CuriosityStream platform, through at least its use of the Akamai CDN, which upon information and belief is a licensed and controlled component of the CuriosityStream platform, determines whether another edge server in the deployment exists from which the requested video content may be obtained.

72. As in claim 18 of the '922 Patent, the Accused Product selects between two or more gateways that are possible sources of a requested data stream by interrogating said possible sources about the loading of the possible source gateways, quality of the data stream and the communication latency between the gateway and the possible source gateways.

How It Works

Simply upload content to a designated online directory using one of many standard upload transport protocols, such as SFTP, or with the higher-performance optional Aspera Upload acceleration feature. NetStorage replicates the content immediately to two or more geographically diverse locations, not only for reliability but also to place the content as close as possible to users. When a user requests content, the Akamai Intelligent Edge Platform automatically selects the copy from the location providing the best performance for a fast, high-quality user experience.

For your online content delivery solution, quality and performance are paramount for end user satisfaction and engagement. NetStorage is designed to operate in concert with Akamai's delivery to specifically provide the highest performance, reliability, and availability ("uptime") possible for your online delivery applications.



1. Customers upload content to the closest NetStorage data center using any of the supported methods: FTP, SFTP, FTPS, SCP, RSYNC, RSYNC over SSH, WGET, NetStorage HTTP API, File Manager (web UI available on the Akamai Luna Control Center), Aspera Upload Acceleration, CMS shell, etc.

2. NetStorage automatically replicates content to the other geographically distributed data centers configured to host that content.

3. Content is delivered to end users worldwide with the best performance thanks to the global Akamai Intelligent Edge Platform.

<https://www.akamai.com/site/en/documents/product-brief/netstorage-product-brief.pdf> (last visited 9/20/2022). Upon information and belief, the CuriosityStream platform, through at least its use of the Akamai CDN, which upon information and belief is a licensed and controlled component of the CuriosityStream platform, selects between two or more edge servers that are possible sources of video content by interrogating possible edge servers about the availability, *e.g.*, loading of the possible edge server, performance, *e.g.*, quality of the data stream, and reliability,

e.g., the communication latency, between the edge server and the other possible source edge servers.

73. Defendant makes, uses, sells, and/or offers to sell the Accused Product which practices at least claim 18 of the '922 Patent.

74. In violation of 35 U.S.C. § 271, Defendant has been directly infringing the '922 Patent, including through its own use, testing, and sale of the Accused Products.

75. Defendant has had knowledge of infringement of the '922 Patent at least as of the service of the present Complaint.

76. Defendant has directly infringed at least one claim of the '922 Patent by making, using, offering for sale, and selling the Accused Product without authority in the United States. As a direct and proximate result of Defendant's direct infringement of the '922 Patent, DMS has been damaged.

77. By engaging in the conduct described herein, Defendant has injured DMS and is thus liable for infringement of the '922 Patent, pursuant to 35 U.S.C. § 271.

78. Defendant has committed these acts of infringement without license or authorization.

79. As a result of Defendant's infringement of the '922 Patent, Plaintiff has suffered monetary damages and is entitled to a monetary judgment in an amount adequate to compensate for Defendant's past infringement, together with interests and costs.

80. Plaintiff reserves the right to modify its infringement theories as discovery progresses in this case; it shall not be estopped for infringement contention or claim construction purposes by the claim chart that it provides with this Complaint.

COUNT III: INFRINGEMENT OF U.S. PATENT NO. 7,739,714

81. Plaintiff incorporates by reference and re-alleges the foregoing paragraphs of this Complaint as if fully set forth herein.

82. Defendant has directly infringed at least claim 9 of the '714 Patent in violation of 35 U.S.C. § 271 *et seq.*, by making, using, offering for sale, or selling in the United States, and/or importing into the United States without authority or license the Accused Product. Further, upon information and belief, Defendant directly infringes at least claim 9 of the '714 Patent through hardware and software systems, such as content delivery systems, that it owns, controls, or licenses.

83. Defendant has directly infringed at least claim 9 of the '714 Patent by implementing a system that includes all the limitations of that claim. For example, claim 9 of the '714 Patent recites:

An object movie processing system for encoding a digital object movie, storing it on a server computer and delivering it to client computers on-line upon request, the system comprising:

a processor and memory;

an encoder for compressing a digital object movie into a series of encoded data blocks stored in a database, each block comprising a sequence of encoded frames;

a storage device that stores the database on a server computer;

a processing unit for accepting a request by a client computer for on-line delivery of the object movie and for determining one or more data blocks to transmit based on a bandwidth associated with the client computer;

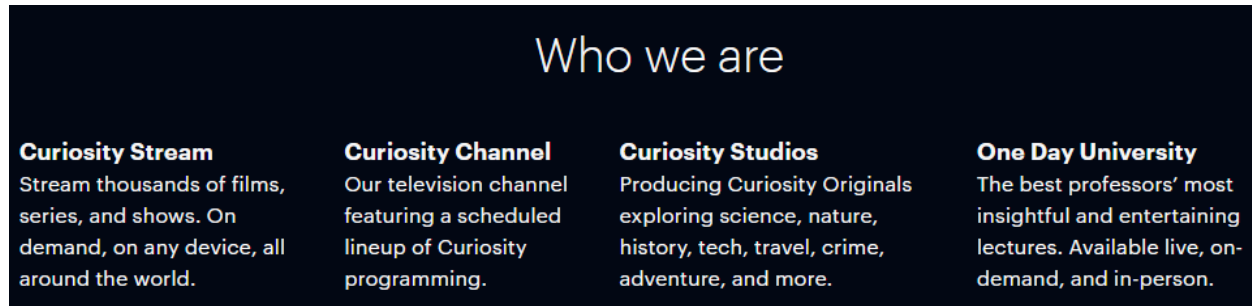
a transmitter for delivering the determined data blocks to the client computer;

a decoder for decompressing the data blocks back into object movie data at the client computer;

an accumulator for ordering the data blocks at the client computer to reconstruct the original digital object movie; and

a player on the client computer for playing the reconstructed digital object movie.

84. As in claim 9 of the ‘714 Patent, the Accused Product is a system for encoding a digital object movie, storing it on a server computer and delivering it to client computers on-line upon request. Upon information and belief, the Defendant has licensed to use and exercises control over its use of the Akamai Content Delivery Network to implement video streaming services.



<https://curiositystream.com/about> (last visited 10/03/2022). As shown, upon information and belief, the CuriosityStream platform, through at least its use of the Akamai CDN, which upon information and belief is a licensed and controlled component of the CuriosityStream platform, along with the CuriosityStream client applications is a system that encodes and stores video content on a server computer for delivery to client computers on-line when requested.

85. As in claim 9 of the ‘714 Patent, the Accused Product, on information and belief, includes a processor and memory. For example, upon information and belief, the CuriosityStream platform, through at least its use of the Akamai CDN, which upon information and belief is a licensed and controlled component of the CuriosityStream platform, includes at least server computers, which include a processor and memory.

86. As in claim 9 of the ‘714 Patent, the Accused Product, on information and belief, includes an encoder for compressing a digital object movie into a series of encoded data blocks stored in a database, each block comprising a sequence of encoded frames. For example, the Accused Product implements MPEG-DASH.

Headers	TextView	SyntaxView	WebForms	HexView	Auth	Cookies	Raw	JSON	XML
<pre>GET https://cdn-s3-akm.curiositystream.com/bitmovin-outputs/WildLogic_HotTuna_ENG_f2997_HD_4227.mp4/fe0a1059ff5885bf85098785bc9f6365.mpd HTTP/1.1 Host: cdn-s3-akm.curiositystream.com Connection: keep-alive sec-ch-ua: "Chromium";v="104", " Not A;Brand";v="99", "Google Chrome";v="104" sec-ch-ua-mobile: ?0 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/104.0.0.0 Safari/537.36 sec-ch-ua-platform: "Windows" Accept: */* Origin: https://curiositystream.com Sec-Fetch-Site: same-site Sec-Fetch-Mode: cors Sec-Fetch-Dest: empty Referer: https://curiositystream.com/ Accept-Encoding: gzip, deflate, br Accept-Language: en-US,en;q=0.9,zh-TW;q=0.8,zh;q=0.7</pre>									
Find... (press Ctrl+Enter to highlight all)								View in Notes	
Transformer	Headers	TextView	SyntaxView	ImageView	HexView	WebView	Auth	Caching	Cookies
Raw	JSON	XML							
<pre><?xml version="1.0" encoding="UTF-8" standalone="yes"?> <MPD id="7b064dfe-67b5-43b8-ae6d-e81a5ec0d613" type="static" xmlns="urn:mpeg:dash:schema:mpd:2011" profiles="urn:mpeg:dash:profile:isoff- live:2011,urn:com:dashif:dash264" xmlns:ns2="http://www.w3.org/1999/xhtml" minBufferTime="POYOMODTOM2.000S" mediaPresentationDuration="POYOMODTOM2M6.724S"><Period id="5cbe54b6-c0fb-48bc-bce0-2b1c6a264685"><AdaptationSet segmentAlignment="true" mimeType="video/mp4"><Representation id="14004476-fadf-4ecb-ba7d-dffdcdc3ef20" width="640" codecs="avc1.640016" height="360" bandwidth="145000" frameRate="30000/1001"><SegmentTemplate media="video/h264/360/145000/base/seg \$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/360/145000/base/init.mp4"/></Representation> <Representation id="1757c519-a8dd-4d3e-885b-3a6858d76f13" width="640" codecs="avc1.640016" height="360" bandwidth="275500" frameRate="30000/1001"><SegmentTemplate media="video/h264/360/275500/base/seg \$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/360/275500/base/init.mp4"/></Representation> <Representation id="bc898879-710c-4149-971f-dcc50faa7f51" width="854" codecs="avc1.640016" height="480" bandwidth="523450" frameRate="30000/1001"><SegmentTemplate media="video/h264/480/523450/base/seg \$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/480/523450/base/init.mp4"/></Representation> <Representation id="320367b1-72e6-4b51-8259-d161f86b6be7" width="960" codecs="avc1.64001F" height="540" bandwidth="821348" frameRate="30000/1001"><SegmentTemplate media="video/h264/540/821348/base/seg \$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/540/821348/base/init.mp4"/></Representation> <Representation id="9496d9f0-6022-4606-8f7e-3d0e27c840a2" width="1280" codecs="avc1.64001F" height="720" bandwidth="1304659" frameRate="30000/1001"><SegmentTemplate media="video/h264/720/1304659/base/seg \$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/720/1304659/base/init.mp4"/></Representation> <Representation id="9193ba9f-b258-4446-b848-dec220f99fd4" width="1280" codecs="avc1.64001F" height="720" bandwidth="2478852" frameRate="30000/1001"><SegmentTemplate media="video/h264/720/2478852/base/seg \$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/720/2478852/base/init.mp4"/></Representation> <Representation id="711cfb60-ff22-4cdb-9d70-b9cf8c16057b" width="1920" codecs="avc1.640028" height="1080" bandwidth="3922908" frameRate="30000/1001"><SegmentTemplate media="video/h264/1080/3922908/base/seg \$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/1080/3922908/base/init.mp4"/></Representation> <Representation id="37ae9eb0-e6fc-475e-9a5f-0901c726bfe2" width="1920" codecs="avc1.640028" height="1080" bandwidth="6208201" frameRate="30000/1001"><SegmentTemplate media="video/h264/1080/6208201/base/seg \$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/1080/6208201/base/init.mp4"/></Representation> </AdaptationSet><AdaptationSet lang="en" label="English" segmentAlignment="true" mimeType="audio/mp4"><Role schemeIdUri="urn:mpeg:dash:role:2011" value="main"/> <Representation id="11d7c54c-5f28-4645-b3fa-76463776dced" codecs="mp4a.40.2" bandwidth="160000" audioSamplingRate="48000"><SegmentTemplate media="audio/full/en/base/seg_ \$Number\$.m4s" duration="192000" timescale="48000" startNumber="0" initialization="audio/full/en/base/init.mp4"/></Representation> schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/> </Representation></AdaptationSet><AdaptationSet mimeType="text/vtt" lang="en" label="English"><Role schemeIdUri="urn:mpeg:dash:role:2011" value="main"/><Representation id="caption_en" bandwidth="256"><BaseURL>subtitles/en/full-1617717983312.vtt</BaseURL> </Representation></AdaptationSet></Period> </MPD></pre>									

As shown, CuriosityStream sends a sequence of encoded frames to a client application.

Scope of MPEG-DASH

Figure 2 illustrates a simple streaming scenario between an HTTP server and a DASH client. In this figure, the multimedia content is captured and stored on an HTTP server and is delivered using HTTP. The content exists on the server in two parts: 1) Media Presentation Description (MPD) which describes a manifest of the available content, its various alternatives, their URL addresses and other characteristics, and 2) Segments which contain the actual multimedia bitstreams in form of chunks, in single or multiple files.

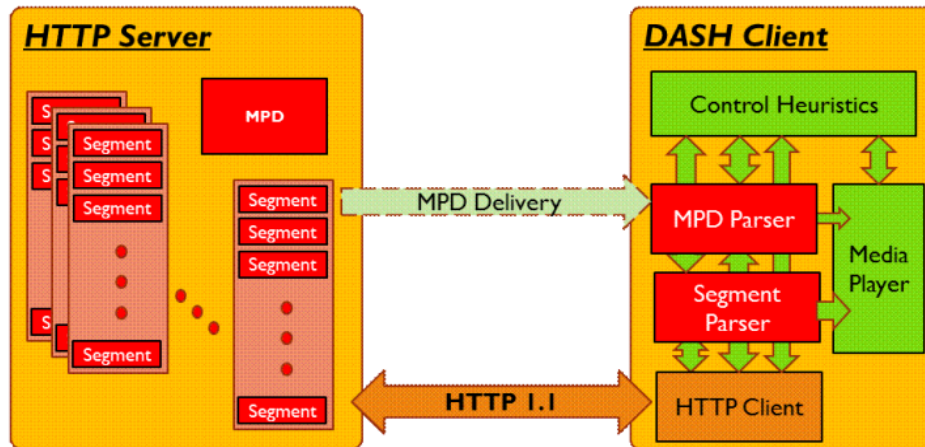


Figure 2. Scope of the MPEG-DASH Standard. The formats and the functionalities of the red blocks are defined by the specification. The clients control heuristics and media players are not within the scope of the standard.

In order to play the content, the DASH client first obtains the MPD. The MPD can be delivered using HTTP, email, thumb drive, broadcast or other transports. By parsing the MPD, the DASH client learns about the timing of the program, the availability of media content, the media types, resolutions, minimum and maximum bandwidths and the existence of various encoded alternatives of multimedia components, the accessibility features and the required digital right management (DRM), the location of each media component on the network and other characteristic of the content. Using this information, the DASH client selects the appropriate encoded alternative and starts streaming of the content by fetching the segments using HTTP GET requests.

https://www.bogotobogo.com/VideoStreaming/images/mpeg_dash/DASH-IEEE-multimedia-

[preprint.pdf](#) (last visited 9/21/2022). Upon information and belief, the CuriosityStream platform, through at least its use of the Akamai CDN, which upon information and belief is a licensed and controlled component of the CuriosityStream platform, uses MPEG-DASH and implements an encoder for compressing video content into a series of encoded data blocks (e.g., segments) stored in a database or equivalent storage system, where each block or segment includes a sequence of encoded frames

87. As in claim 9 of the '714 Patent, the Accused Product, on information and belief, includes a storage device that stores the database on a server computer.

Figure 1 shows a possible deployment architecture in which the formats defined in this document may be used. Boxes with solid lines indicate devices that are mentioned in this document as they host or process the formats defined in this document whereas dashed boxes are conceptual or transparent. This document deals with the definition of formats that are accessible on the interface to the DASH Client, indicated by the solid lines. Any other formats or interfaces are outside the scope of this document. In the considered deployment scenario, it is assumed that the DASH Client has access to an MPD. The MPD provides sufficient information for the DASH Client to provide a streaming service to the user by requesting Segments from an HTTP server and demultiplexing, decoding and rendering the included media streams.

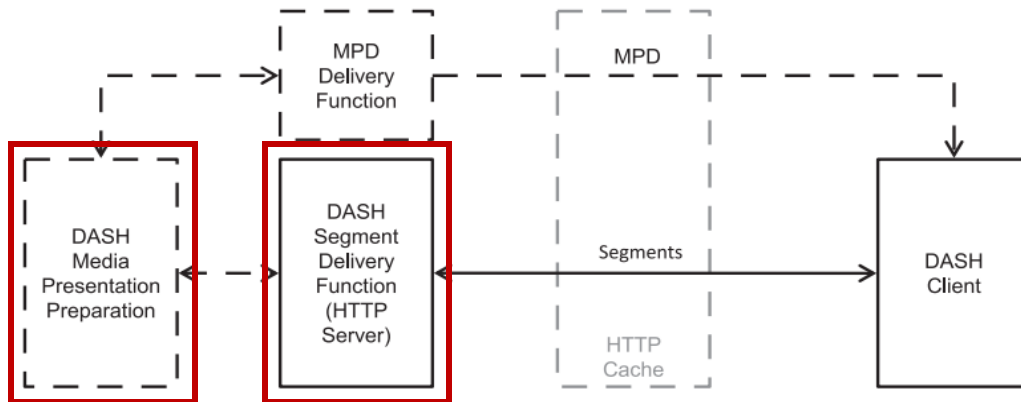


Figure 1 — Example system for DASH formats

Although the formats are initially designed to be used in the above deployment scenario, their application is obviously not restricted to this scenario. The particular aspect on "HTTP" in DASH is the usage of HTTP-URLs in the MPD for the purpose to refer to Segments. The usage of HTTP-URLs enables unique location information and it provides well-defined methods to access the resources, in particular HTTP GET and HTTP partial GET.

ISO/IEC 23009-1:2019. As shown, MPEG-DASH suggests a central repository for DASH Media that is retrieved for delivery, for example, via HTTP. Upon information and belief, the CuriosityStream platform, through at least its use of the Akamai CDN, which upon information and belief is a licensed and controlled component of the CuriosityStream platform, includes a storage device that stores the database or equivalent storage system on a server computer.

88. As in claim 9 of the '714 Patent, the Accused Product, on information and belief, includes a processing unit for accepting a request by a client computer for on-line delivery of the object movie and for determining one or more data blocks to transmit based on a bandwidth associated with the client computer.

Figure 1 shows a possible deployment architecture in which the formats defined in this document may be used. Boxes with solid lines indicate devices that are mentioned in this document as they host or process the formats defined in this document whereas dashed boxes are conceptual or transparent. This document deals with the definition of formats that are accessible on the interface to the DASH Client, indicated by the solid lines. Any other formats or interfaces are outside the scope of this document. In the considered deployment scenario, it is assumed that the DASH Client has access to an MPD. The MPD provides sufficient information for the DASH Client to provide a streaming service to the user by requesting Segments from an HTTP server and demultiplexing, decoding and rendering the included media streams.

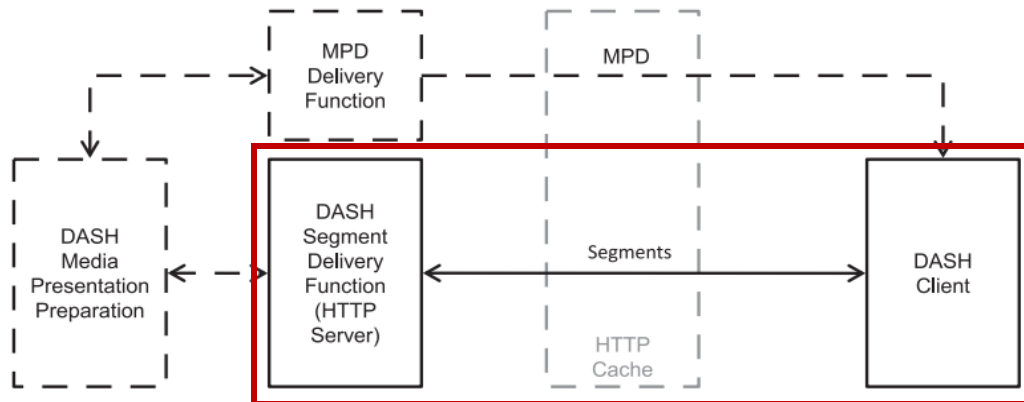


Figure 1 — Example system for DASH formats

Although the formats are initially designed to be used in the above deployment scenario, their application is obviously not restricted to this scenario. The particular aspect on "HTTP" in DASH is the usage of HTTP-URLs in the MPD for the purpose to refer to Segments. The usage of HTTP-URLs enables unique location information and it provides well-defined methods to access the resources, in particular HTTP GET and HTTP partial GET.

4.2 DASH Client model

The design of the formats defined in this document is based on the informative client model as shown in Figure 2. The figure illustrates the logical components of a conceptual DASH Client model and the relation to other components in a media streaming application. In this figure, the DASH access engine receives the Media Presentation Description (MPD), constructs and issues requests and receives Segments or parts of Segments. The DASH Client may use metadata provided in the MPD for the selection of media components by communication with the media streaming application. Such metadata may for example include codec capability information, language codes, accessibility information and other information for the selection of media components. In the context of this document, the output of the DASH access engine consists of media in MPEG container formats (ISO/IEC 14496-12 ISO base media file format or ISO/IEC 13818-1 MPEG-2 Transport Stream), or parts thereof, together with timing information that maps the internal timing of the continuous media to the timeline of the Media Presentation. In Annex F, guidance on enabling the use of this document with other container formats is provided. In addition, the DASH access client may also receive and extract Events that are related to the media time. The events may be processed in the DASH Client or may be forwarded to an event processing application in the execution environment of the DASH Client.

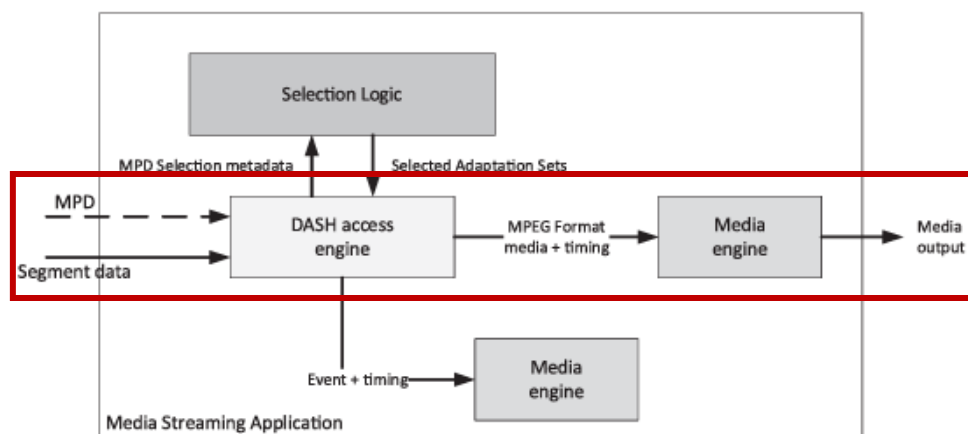


Figure 2 — DASH Client model

ISO/IEC 23009-1:2019.

Scope of MPEG-DASH

Figure 2 illustrates a simple streaming scenario between an HTTP server and a DASH client. In this figure, the multimedia content is captured and stored on an HTTP server and is delivered using HTTP. The content exists on the server in two parts: 1) Media Presentation Description (MPD) which describes a manifest of the available content, its various alternatives, their URL addresses and other characteristics, and 2) Segments which contain the actual multimedia bitstreams in form of chunks, in single or multiple files.

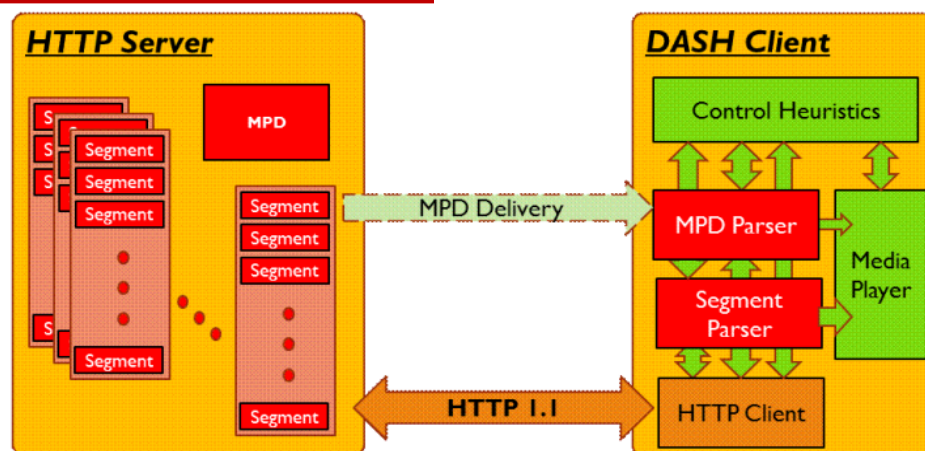


Figure 2. Scope of the MPEG-DASH Standard. The formats and the functionalities of the red blocks are defined by the specification. The clients control heuristics and media players are not within the scope of the standard.

In order to play the content, the DASH client first obtains the MPD. The MPD can be delivered using HTTP, email, thumb drive, broadcast or other transports. By parsing the MPD, the DASH client learns about the timing of the program, the availability of media content, the media types, resolutions, minimum and maximum bandwidths and the existence of various encoded alternatives of multimedia components, the accessibility features and the required digital right management (DRM), the location of each media component on the network and other characteristic of the content. Using this information, the DASH client selects the appropriate encoded alternative and starts streaming of the content by fetching the segments using HTTP GET requests.

https://www.bogotobogo.com/VideoStreaming/images/mpeg_dash/DASH-IEEE-multimedia-preprint.pdf (last visited 9/21/2022).

Headers	TextView	SyntaxView	WebForms	HexView	Auth	Cookies	Raw	JSON	XML						
<pre>GET https://cdn-s3-akm.curiositystream.com/bitmovin-outputs/WildLogic_HotTuna_ENG_f2997_HD_4227.mp4/fe0a1059ff5885bf85098785bc9f6365.mpd HTTP/1.1 Host: cdn-s3-akm.curiositystream.com Connection: keep-alive sec-ch-ua: "Chromium";v="104", " Not A;Brand";v="99", "Google Chrome";v="104" sec-ch-ua-mobile: ?0 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/104.0.0.0 Safari/537.36 sec-ch-ua-platform: "Windows" Accept: */* Origin: https://curiositystream.com Sec-Fetch-Site: same-site Sec-Fetch-Mode: cors Sec-Fetch-Dest: empty Referer: https://curiositystream.com/ Accept-Encoding: gzip, deflate, br Accept-Language: en-US,en;q=0.9,zh-TW;q=0.8,zh;q=0.7</pre>															
Find... (press Ctrl+Enter to highlight all) View in Notes															
Transformer	Headers	TextView	SyntaxView	ImageView	HexView	WebView	Auth	Caching	Cookies						
<table border="1"> <thead> <tr> <th>Raw</th> <th>JSON</th> <th>XML</th> </tr> </thead> <tbody> <tr> <td colspan="3"> <pre><?xml version="1.0" encoding="UTF-8" standalone="yes"?> <MPD id="7b064dfe-67b5-43b8-ae6d-e81a5ec0d613" type="static" xmlns="urn:mpeg:dash:schema:mpd:2011" profiles="urn:mpeg:dash:profile:isoff- live:2011,urn:com:dashif:dash264" xmlns:ns2="http://www.w3.org/1999/xlink" minBufferTime="POYOMODT0H0M2.000S" mediaPresentationDuration="POYOMODT0H52M6.724S"><Period id="5cbe54b6-c0fb-48bc-bce0-2b1c6a264685"><AdaptationSet segmentAlignment="true" mimeType="video/mp4"><Representation id="14004476-fadf-4ecb-ba7d-dffdcdc3ef20" width="640" codecs="avc1.640016" height="360" bandwidth="145000" frameRate="30000/1001"><SegmentTemplate media="video/h264/360/145000/base/seg_\$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/360/145000/base/init.mp4"/></Representation> <Representation id="1757c519-a8dd-4d3e-885b-3a6858d76f13" width="640" codecs="avc1.640016" height="360" bandwidth="275500" frameRate="30000/1001"><SegmentTemplate media="video/h264/360/275500/base/seg_\$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/360/275500/base/init.mp4"/></Representation> <Representation id="bc898879-710c-4149-971f-dcc50faa7f51" width="854" codecs="avc1.640016" height="480" bandwidth="523450" frameRate="30000/1001"><SegmentTemplate media="video/h264/480/523450/base/seg_\$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/480/523450/base/init.mp4"/></Representation> <Representation id="320367b1-72e6-4b51-8259-d161f86b6be7" width="960" codecs="avc1.64001F"</pre> </td> </tr> </tbody> </table>										Raw	JSON	XML	<pre><?xml version="1.0" encoding="UTF-8" standalone="yes"?> <MPD id="7b064dfe-67b5-43b8-ae6d-e81a5ec0d613" type="static" xmlns="urn:mpeg:dash:schema:mpd:2011" profiles="urn:mpeg:dash:profile:isoff- live:2011,urn:com:dashif:dash264" xmlns:ns2="http://www.w3.org/1999/xlink" minBufferTime="POYOMODT0H0M2.000S" mediaPresentationDuration="POYOMODT0H52M6.724S"><Period id="5cbe54b6-c0fb-48bc-bce0-2b1c6a264685"><AdaptationSet segmentAlignment="true" mimeType="video/mp4"><Representation id="14004476-fadf-4ecb-ba7d-dffdcdc3ef20" width="640" codecs="avc1.640016" height="360" bandwidth="145000" frameRate="30000/1001"><SegmentTemplate media="video/h264/360/145000/base/seg_\$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/360/145000/base/init.mp4"/></Representation> <Representation id="1757c519-a8dd-4d3e-885b-3a6858d76f13" width="640" codecs="avc1.640016" height="360" bandwidth="275500" frameRate="30000/1001"><SegmentTemplate media="video/h264/360/275500/base/seg_\$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/360/275500/base/init.mp4"/></Representation> <Representation id="bc898879-710c-4149-971f-dcc50faa7f51" width="854" codecs="avc1.640016" height="480" bandwidth="523450" frameRate="30000/1001"><SegmentTemplate media="video/h264/480/523450/base/seg_\$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/480/523450/base/init.mp4"/></Representation> <Representation id="320367b1-72e6-4b51-8259-d161f86b6be7" width="960" codecs="avc1.64001F"</pre>		
Raw	JSON	XML													
<pre><?xml version="1.0" encoding="UTF-8" standalone="yes"?> <MPD id="7b064dfe-67b5-43b8-ae6d-e81a5ec0d613" type="static" xmlns="urn:mpeg:dash:schema:mpd:2011" profiles="urn:mpeg:dash:profile:isoff- live:2011,urn:com:dashif:dash264" xmlns:ns2="http://www.w3.org/1999/xlink" minBufferTime="POYOMODT0H0M2.000S" mediaPresentationDuration="POYOMODT0H52M6.724S"><Period id="5cbe54b6-c0fb-48bc-bce0-2b1c6a264685"><AdaptationSet segmentAlignment="true" mimeType="video/mp4"><Representation id="14004476-fadf-4ecb-ba7d-dffdcdc3ef20" width="640" codecs="avc1.640016" height="360" bandwidth="145000" frameRate="30000/1001"><SegmentTemplate media="video/h264/360/145000/base/seg_\$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/360/145000/base/init.mp4"/></Representation> <Representation id="1757c519-a8dd-4d3e-885b-3a6858d76f13" width="640" codecs="avc1.640016" height="360" bandwidth="275500" frameRate="30000/1001"><SegmentTemplate media="video/h264/360/275500/base/seg_\$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/360/275500/base/init.mp4"/></Representation> <Representation id="bc898879-710c-4149-971f-dcc50faa7f51" width="854" codecs="avc1.640016" height="480" bandwidth="523450" frameRate="30000/1001"><SegmentTemplate media="video/h264/480/523450/base/seg_\$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/480/523450/base/init.mp4"/></Representation> <Representation id="320367b1-72e6-4b51-8259-d161f86b6be7" width="960" codecs="avc1.64001F"</pre>															

Upon information and belief, the CuriosityStream platform, through at least its use of the Akamai CDN, which upon information and belief is a licensed and controlled component of the CuriosityStream platform, along with the CuriosityStream client applications have an implementation of MPEG-DASH which includes a server-side architecture that accepts a request by a user's computer for on-line delivery of video content and determines one or more data blocks or segments to transmit based on a bandwidth associated with the user's computer.

89. As in claim 9 of the '714 Patent, the Accused Product, on information and belief, includes a transmitter for delivering the determined data blocks to the client computer.

Figure 1 shows a possible deployment architecture in which the formats defined in this document may be used. Boxes with solid lines indicate devices that are mentioned in this document as they host or process the formats defined in this document whereas dashed boxes are conceptual or transparent. This document deals with the definition of formats that are accessible on the interface to the DASH Client, indicated by the solid lines. Any other formats or interfaces are outside the scope of this document. In the considered deployment scenario, it is assumed that the DASH Client has access to an MPD. The MPD provides sufficient information for the DASH Client to provide a streaming service to the user by requesting Segments from an HTTP server and demultiplexing, decoding and rendering the included media streams.

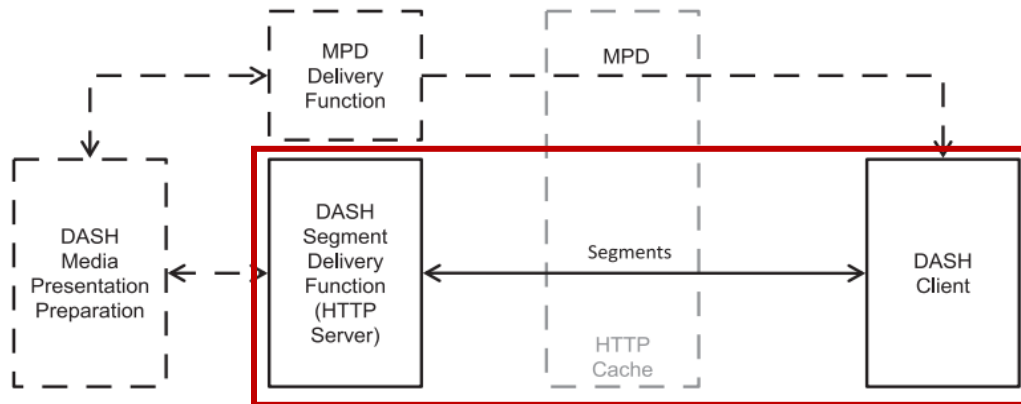


Figure 1 — Example system for DASH formats

Although the formats are initially designed to be used in the above deployment scenario, their application is obviously not restricted to this scenario. The particular aspect on "HTTP" in DASH is the usage of HTTP-URLs in the MPD for the purpose to refer to Segments. The usage of HTTP-URLs enables unique location information and it provides well-defined methods to access the resources, in particular HTTP GET and HTTP partial GET.

ISO/IEC 23009-1:2019.

Headers	TextView	SyntaxView	WebForms	HexView	Auth	Cookies	Raw	JSON	XML
<pre>GET https://cdn-s3-akm.curiositystream.com/bitmovin-outputs/WildLogic_HotTuna_ENG_f2997_HD_4227.mp4/fe0a1059ff5885bf85098785bc9f6365.mpd HTTP/1.1 Host: cdn-s3-akm.curiositystream.com Connection: keep-alive sec-ch-ua: "Chromium";v="104", " Not A;Brand";v="99", "Google Chrome";v="104" sec-ch-ua-mobile: ?0 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/104.0.0.0 Safari/537.36 sec-ch-ua-platform: "Windows" Accept: */* Origin: https://curiositystream.com Sec-Fetch-Site: same-site Sec-Fetch-Mode: cors Sec-Fetch-Dest: empty Referer: https://curiositystream.com/ Accept-Encoding: gzip, deflate, br Accept-Language: en-US,en;q=0.9,zh-TW;q=0.8,zh;q=0.7</pre>									
Find... (press Ctrl+Enter to highlight all) View in Notes									
Transformer	Headers	TextView	SyntaxView	ImageView	HexView	WebView	Auth	Caching	Cookies
Raw	JSON	XML							
<pre><?xml version="1.0" encoding="UTF-8" standalone="yes"?> <MPD id="7b064dfe-67b5-43b8-ae6d-e81a5ec0d613" type="static" xmlns="urn:mpeg:dash:schema:mpd:2011" profiles="urn:mpeg:dash:profile:isoff- live:2011,urn:com:dashif:dash264" xmlns:ns2="http://www.w3.org/1999/xlink" minBufferTime="POYOMODTOM2.000S" mediaPresentationDuration="POYOMODTOM2M6.724S"><Period id="5cbe54b6-c0fb-48bc-bce0-2b1c6a264685"><AdaptationSet segmentAlignment="true" mimeType="video/mp4"><Representation id="14004476-fadf-4ecb-ba7d-dffdc3ef20" width="640" codecs="avc1.640016" height="360" bandwidth="145000" frameRate="30000/1001"><SegmentTemplate media="video/h264/360/145000/base/seg \$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/360/145000/base/init.mp4"/></Representation> <Representation id="1757c519-a8dd-4d3e-885b-3a6858d76f13" width="640" codecs="avc1.640016" height="360" bandwidth="275500" frameRate="30000/1001"><SegmentTemplate media="video/h264/360/275500/base/seg \$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/360/275500/base/init.mp4"/></Representation> <Representation id="bc898879-710c-4149-971f-dcc50faa7f51" width="854" codecs="avc1.640016" height="480" bandwidth="523450" frameRate="30000/1001"><SegmentTemplate media="video/h264/480/523450/base/seg \$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/480/523450/base/init.mp4"/></Representation> <Representation id="320367b1-72e6-4b51-8259-d161f86b6be7" width="960" codecs="avc1.64001F" height="540" bandwidth="821348" frameRate="30000/1001"><SegmentTemplate media="video/h264/540/821348/base/seg \$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/540/821348/base/init.mp4"/></Representation> <Representation id="9496d9f0-6022-4606-8f7e-3d0e27c840a2" width="1280" codecs="avc1.64001F" height="720" bandwidth="1304659" frameRate="30000/1001"><SegmentTemplate media="video/h264/720/1304659/base/seg \$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/720/1304659/base/init.mp4"/></Representation> <Representation id="9193ba9f-b258-4446-b848-dec220f99fd4" width="1280" codecs="avc1.64001F" height="720" bandwidth="2478852" frameRate="30000/1001"><SegmentTemplate media="video/h264/720/2478852/base/seg \$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/720/2478852/base/init.mp4"/></Representation> <Representation id="711cfb60-ff22-4cdb-9d70-b9cf8c16057b" width="1920" codecs="avc1.640028" height="1080" bandwidth="3922908" frameRate="30000/1001"><SegmentTemplate media="video/h264/1080/3922908/base/seg \$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/1080/3922908/base/init.mp4"/></Representation> <Representation id="37ae9eb0-e6fc-475e-9a5f-0901c726bfe2" width="1920" codecs="avc1.640028" height="1080" bandwidth="6208201" frameRate="30000/1001"><SegmentTemplate media="video/h264/1080/6208201/base/seg \$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/1080/6208201/base/init.mp4"/></Representation> </AdaptationSet><AdaptationSet lang="en" label="English" segmentAlignment="true" mimeType="audio/mp4"><Role schemeIdUri="urn:mpeg:dash:role:2011" value="main"/> <Representation id="11d7c54c-5f28-4645-b3fa-76463776dced" codecs="mp4a.40.2" bandwidth="160000" audioSamplingRate="48000"><SegmentTemplate media="audio/full/en/base/seg_ \$Number\$.m4s" duration="192000" timescale="48000" startNumber="0" initialization="audio/full/en/base/init.mp4"/></Representation> </AdaptationSet><AdaptationSet mimeType="text/vtt" lang="en" label="English"><Role schemeIdUri="urn:mpeg:dash:role:2011" value="main"/><Representation id="caption_en" bandwidth="256"><BaseURL>subtitles/en/full-1617717983312.vtt</BaseURL> </Representation></AdaptationSet></Period> </MPD></pre>									

As shown, CuriosityStream sends blocks of data to the client application. Upon information and belief, the CuriosityStream platform, through at least its use of the Akamai CDN, which upon information and belief is a licensed and controlled component of the CuriosityStream platform, includes a transmitter for sending the data blocks to the client computer.

90. As in claim 9 of the '714 Patent, the Accused Product, on information and belief, includes a decoder for decompressing the data blocks back into object movie data at the client computer.

Figure 1 shows a possible deployment architecture in which the formats defined in this document may be used. Boxes with solid lines indicate devices that are mentioned in this document as they host or process the formats defined in this document whereas dashed boxes are conceptual or transparent. This document deals with the definition of formats that are accessible on the interface to the DASH Client, indicated by the solid lines. Any other formats or interfaces are outside the scope of this document. In the considered deployment scenario, it is assumed that the DASH Client has access to an MPD. The MPD provides sufficient information for the DASH Client to provide a streaming service to the user by requesting Segments from an HTTP server and demultiplexing, decoding and rendering the included media streams.

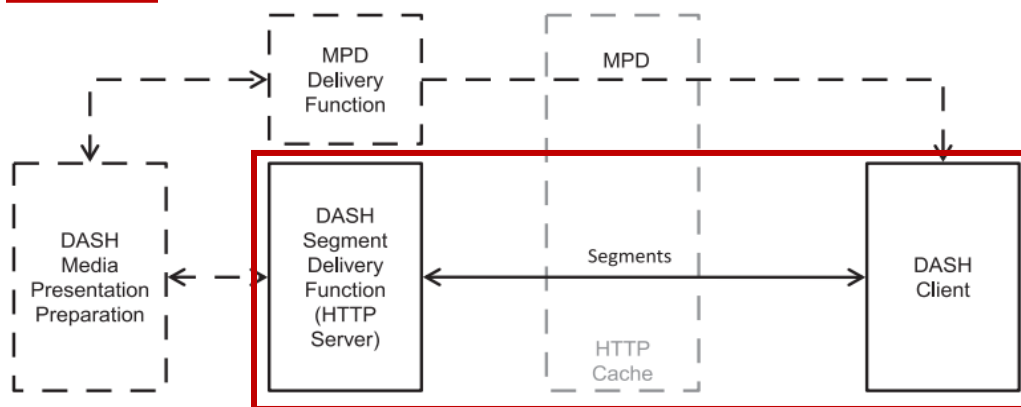
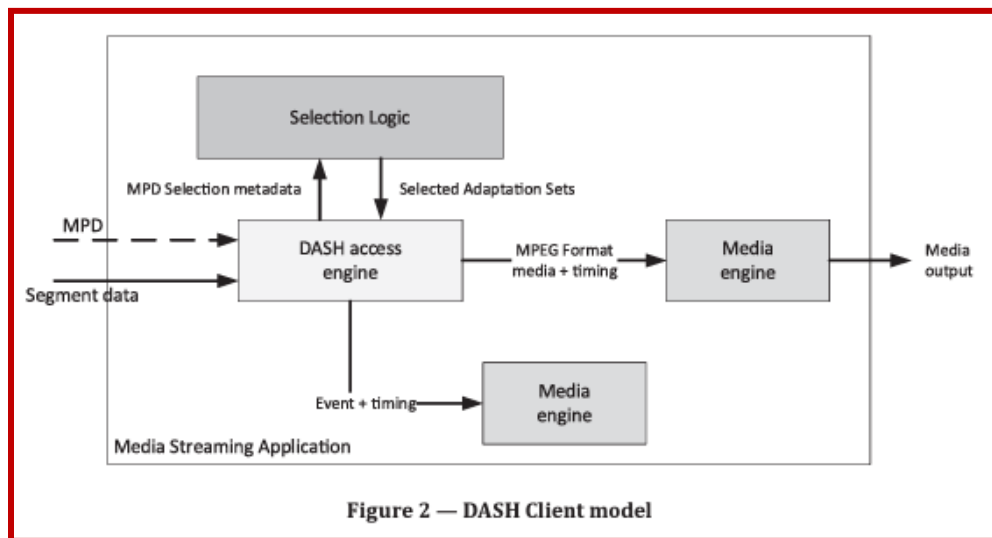


Figure 1 — Example system for DASH formats

Although the formats are initially designed to be used in the above deployment scenario, their application is obviously not restricted to this scenario. The particular aspect on "HTTP" in DASH is the usage of HTTP-URLs in the MPD for the purpose to refer to Segments. The usage of HTTP-URLs enables unique location information and it provides well-defined methods to access the resources, in particular HTTP GET and HTTP partial GET.

4.2 DASH Client model

The design of the formats defined in this document is based on the informative client model as shown in [Figure 2](#). The figure illustrates the logical components of a conceptual DASH Client model and the relation to other components in a media streaming application. In this figure, the DASH access engine receives the Media Presentation Description (MPD), constructs and issues requests and receives Segments or parts of Segments. The DASH Client may use metadata provided in the MPD for the selection of media components by communication with the media streaming application. Such metadata may for example include codec capability information, language codes, accessibility information and other information for the selection of media components. In the context of this document, the output of the DASH access engine consists of media in MPEG container formats (ISO/IEC 14496-12 ISO base media file format or ISO/IEC 13818-1 MPEG-2 Transport Stream), or parts thereof, together with timing information that maps the internal timing of the continuous media to the timeline of the Media Presentation. In [Annex F](#), guidance on enabling the use of this document with other container formats is provided. In addition, the DASH access client may also receive and extract Events that are related to the media time. The events may be processed in the DASH Client or may be forwarded to an event processing application in the execution environment of the DASH Client.



ISO/IEC 23009-1:2019.

Scope of MPEG-DASH

Figure 2 illustrates a simple streaming scenario between an HTTP server and a DASH client. In this figure, the multimedia content is captured and stored on an HTTP server and is delivered using HTTP. The content exists on the server in two parts: 1) Media Presentation Description (MPD) which describes a manifest of the available content, its various alternatives, their URL addresses and other characteristics, and 2) Segments which contain the actual multimedia bitstreams in form of chunks, in single or multiple files.

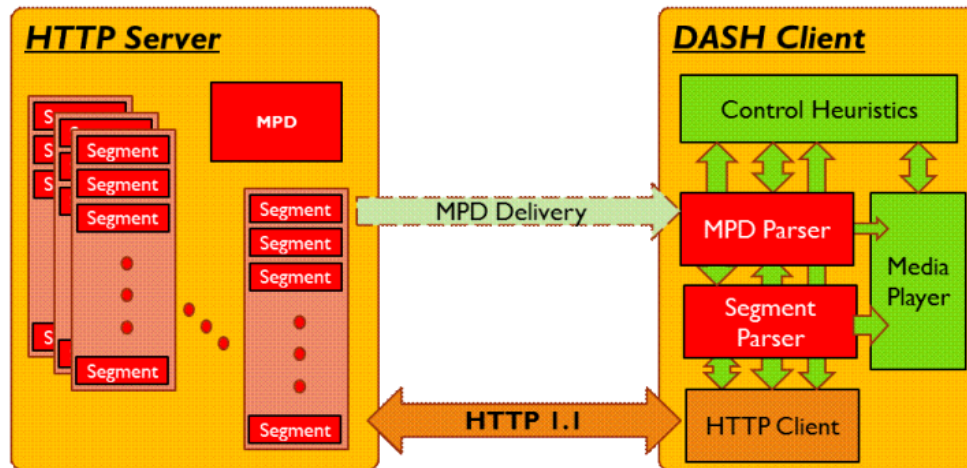


Figure 2. Scope of the MPEG-DASH Standard. The formats and the functionalities of the red blocks are defined by the specification. The clients control heuristics and media players are not within the scope of the standard.

In order to play the content, the DASH client first obtains the MPD. The MPD can be delivered using HTTP, email, thumb drive, broadcast or other transports. By parsing the MPD, the DASH client learns about the timing of the program, the availability of media content, the media types, resolutions, minimum and maximum bandwidths and the existence of various encoded alternatives of multimedia components, the accessibility features and the required digital right management (DRM), the location of each media component on the network and other characteristic of the content. Using this information, the DASH client selects the appropriate encoded alternative and starts streaming of the content by fetching the segments using HTTP GET requests.

After appropriate buffering to allow for network throughput variations, the client continues fetching the subsequent segments and also monitors the bandwidth fluctuations of the network. Depending on its measurements, the client decides how to adapt to the available bandwidth by fetching segments of different alternatives (with lower or higher bitrate) to maintain an adequate buffer.

The MPEG-DASH specification only defines the MPD and the segment formats. The delivery of the MPD and the media encoding formats containing the segments as well as the client behavior for fetching, adaptation heuristics and playing the content are outside of MPEG-DASH's scope.

https://www.bogotobogo.com/VideoStreaming/images/mpeg_dash/DASH-IEEE-multimedia-

[preprint.pdf](#) (last visited 9/21/2022). Upon information and belief, the CuriosityStream platform,

through its client software implementations, includes a decoder for decompressing the data blocks back into video content at the client computer.

91. As in claim 9 of the '714 Patent, the Accused Product, on information and belief, includes an accumulator for ordering the data blocks at the client computer to reconstruct the original digital object movie.

#	Result	Protocol	Host	URL
94	200	HTTPS	cdn-s3-akm.curiositystre...	/bitmovin-outputs/WildLogic_Ho...
95	200	HTTPS	cdn-s3-akm.curiositystre...	/bitmovin-outputs/WildLogic_Ho...
96	200	HTTPS	cdn-s3-akm.curiositystre...	/bitmovin-outputs/WildLogic_Ho...
97	204	HTTPS	m.clarity.ms	/collect
98	200	HTTPS	www.facebook.com	/tr/?id=1418728098427502&ev...
99	200	HTTPS	cdn.curiositystream.com	/webapp-v3/images/v8/video-3...

As shown, CuriosityStream sends data blocks of video data to the computer to be reconstructed into the selected video content.

Scope of MPEG-DASH

Figure 2 illustrates a simple streaming scenario between an HTTP server and a DASH client. In this figure, the multimedia content is captured and stored on an HTTP server and is delivered using HTTP. The content exists on the server in two parts: 1) Media Presentation Description (MPD) which describes a manifest of the available content, its various alternatives, their URL addresses and other characteristics, and 2) Segments which contain the actual multimedia bitstreams in form of chunks, in single or multiple files.

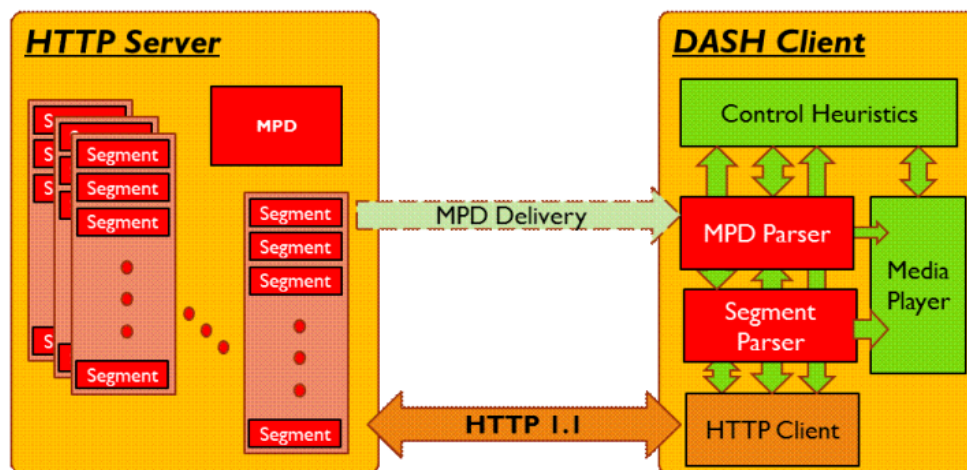


Figure 2. Scope of the MPEG-DASH Standard. The formats and the functionalities of the red blocks are defined by the specification. The clients control heuristics and media players are not within the scope of the standard.

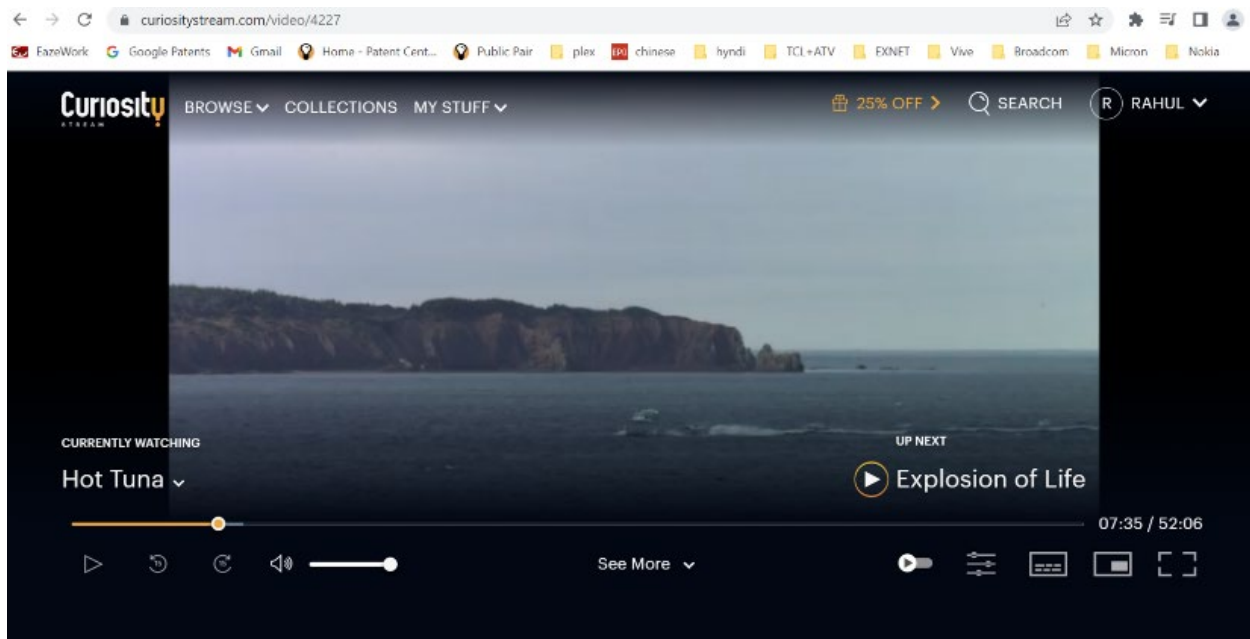
In order to play the content, the DASH client first obtains the MPD. The MPD can be delivered using HTTP, email, thumb drive, broadcast or other transports. By parsing the MPD, the DASH client learns about the timing of the program, the availability of media content, the media types, resolutions, minimum and maximum bandwidths and the existence of various encoded alternatives of multimedia components, the accessibility features and the required digital right management (DRM), the location of each media component on the network and other characteristic of the content. Using this information, the DASH client selects the appropriate encoded alternative and starts streaming of the content by fetching the segments using HTTP GET requests.

After appropriate buffering to allow for network throughput variations, the client continues fetching the subsequent segments and also monitors the bandwidth fluctuations of the network. Depending on its measurements, the client decides how to adapt to the available bandwidth by fetching segments of different alternatives (with lower or higher bitrate) to maintain an adequate buffer.

The MPEG-DASH specification only defines the MPD and the segment formats. The delivery of the MPD and the media encoding formats containing the segments as well as the client behavior for fetching, adaptation heuristics and playing the content are outside of MPEG-DASH's scope.

https://www.bogotobogo.com/VideoStreaming/images/mpeg_dash/DASH-IEEE-multimedia-preprint.pdf (last visited 9/21/2022). Upon information and belief, the CuriosityStream platform, through its client software implementations, includes an accumulator for ordering the data blocks at the client computer to reconstruct the original video content.

92. As in claim 9 of the '714 Patent, the Accused Product includes a player on the client computer for playing the reconstructed digital object movie.



<https://curiositystream.com/video/4227> (last visited 10/06/2022). Upon information and belief, the CuriosityStream platform, through its client software implementations, includes a video playback implementation on the client computer for playing the reconstructed video content.

93. Defendant makes, uses, sells, and/or offers to sell the Accused Product which practices at least claim 9 of the '714 Patent.

94. In violation of 35 U.S.C. § 271, Defendant has been directly infringing the '714 Patent, including through its own use, testing, and sale of the Accused Products.

95. Defendant has had knowledge of infringement of the '714 Patent at least as of the service of the present Complaint.

96. Defendant has directly infringed at least one claim of the '714 Patent by making, using, offering for sale, and selling the Accused Product without authority in the United States. As a direct and proximate result of Defendant's direct infringement of the '714 Patent, DMS has been damaged.

97. By engaging in the conduct described herein, Defendant has injured DMS and is thus liable for infringement of the '714 Patent, pursuant to 35 U.S.C. § 271.

98. Defendant has committed these acts of infringement without license or authorization.

99. As a result of Defendant's infringement of the '714 Patent, Plaintiff has suffered monetary damages and is entitled to a monetary judgment in an amount adequate to compensate for Defendant's past infringement, together with interests and costs.

100. Plaintiff reserves the right to modify its infringement theories as discovery progresses in this case; it shall not be estopped for infringement contention or claim construction purposes by the claim chart that it provides with this Complaint.

COUNT IV: INFRINGEMENT OF U.S. PATENT NO. 8,122,004

101. Plaintiff incorporates by reference and re-alleges the foregoing paragraphs of this Complaint as if fully set forth herein.

102. Defendant has directly infringed at least claim 1 of the '004 Patent in violation of 35 U.S.C. § 271 *et seq.*, by making, using, offering for sale, or selling in the United States, and/or importing into the United States without authority or license the Accused Product. Further, upon information and belief, Defendant directly infringes at least claim 1 of the '004 Patent through

hardware and software systems, such as content delivery systems, that it owns, controls, or licenses.

103. The Accused Product meets all the limitations of at least claim 1 of the '004 Patent.

For example, claim 1 of the '004 Patent recites:

A computer-implemented method implemented by at least one device of a content delivery system, the method comprising:

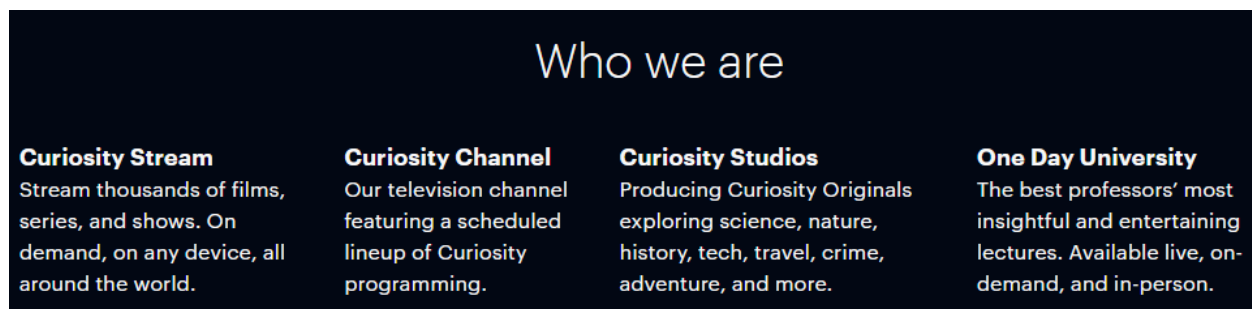
processing requests for media content from network devices to determine whether the network devices are supported by the content delivery system; and for at least one network device that is supported:

configuring a content package for media content that is requested by the network device to include code specific to a browser of the network device identified based upon one or more attributes of the network device;

causing communication of the content package to the network device; and

wherein the one or more attributes include one or more of basic operating characteristics of the at least one network device, a language attribute, a bandwidth attribute, a firewall attribute, or a permissions attribute.

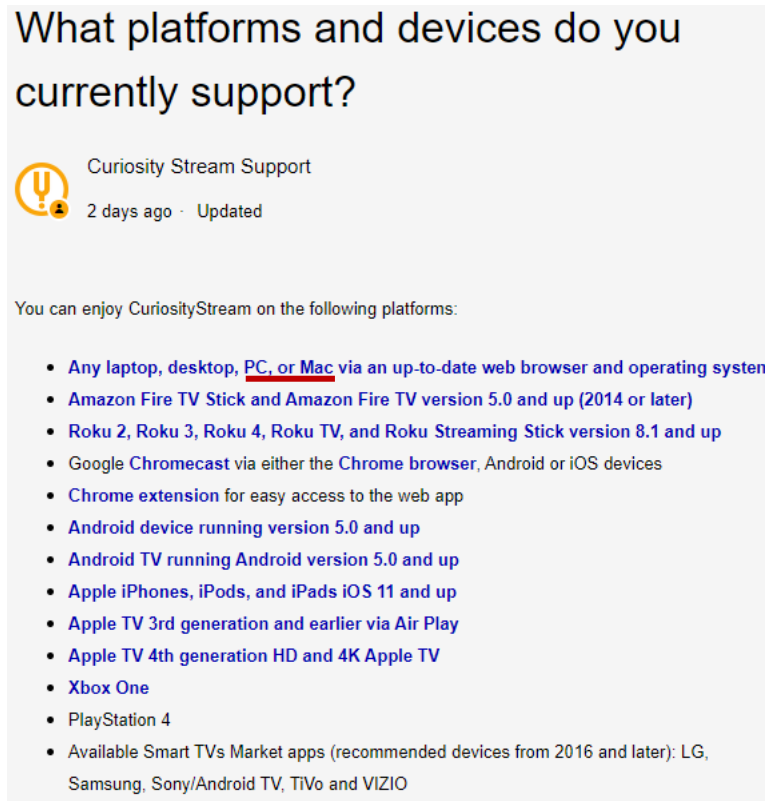
104. As in claim 1 of the '004 Patent, the Accused Product provides a content (*e.g.*, video content) delivery system. Upon information and belief, the Defendant has licensed to use and exercises control over its use of the Akamai Content Delivery Network to implement video streaming services.



<https://curiositystream.com/about> (last visited 10/03/2022).

105. As in claim 1 of the '004 Patent, the Accused Product processes requests for media content from network devices (*e.g.*, computers) to determine whether the network devices are

supported by the content delivery system (*e.g.*, Curiosity). For example, the Accused Product provides functionality to allow the streaming of media content to the browser of certain computer operating systems.



<https://help.curiositystream.com/hc/en-us/articles/209260738-What-platforms-and-devices-do-you-currently-support-> (last visited 10/06/2022). As shown, upon information and belief, the

CuriosityStream platform, through at least its use of the Akamai CDN, which upon information and belief is a licensed and controlled component of the CuriosityStream platform, along with the CuriosityStream client applications provides functionality to allow the streaming of video content to the browser of certain computer operating systems, *e.g.*, Mac and PC.

106. As in claim 1 of the '004 Patent, the Accused Product configures a content package for media content (*e.g.*, video content) that is requested by the network device (*e.g.*, a computer) to include code specific to a browser of the network device identified based upon one or more

attributes of the network device. For example, the Accused Product provides the media content to the browser based on certain specific attributes, such as language and browser type.

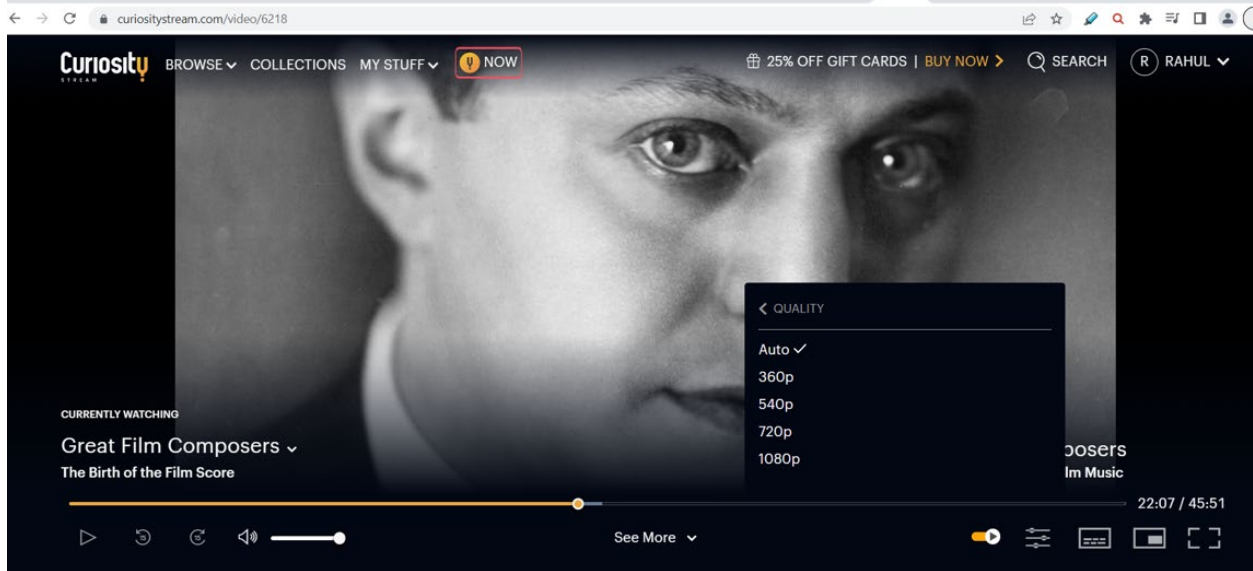
```

GET https://api.curiositystream.com/v1/media/6218?encodingsNew=true&encodingsFormat=mpd HTTP/1.1
Host: api.curiositystream.com
Connection: keep-alive
Sec-ch-ua: "Chromium";v="104", " Not A:Brand";v="99", "Google Chrome";v="104"
Accept-Language: en-US
Sec-ch-ua-mobile: ?0
x-platform: web
x-client-version: webapp-1.26.18
Content-Type: application/json
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/104.0.0.0 Safari/537.36
x-auth-token: Account eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJodHRwczp1wVYXBPbmN1cm1vc210eXN0cmVhbS5jb21cL3YxXC9sb2dpbiIsIm1hdCIGMTY2MDgyOTY4NSwiZXBwIjoxNjkyMzg3Mjg1LCJ1b2Y1OiJlZ2NjA4Mjk2ODUsImp0aSI6Ing5c2ROVUVzVXJrb21mTmE1LCJzZW50Ij1iZWE3NDhmZC03NTQxLTQwOWYtyjIzZSoyZWZiNGRjNGM3ZDgiLCJwcnYiOiIwZjMwZmRlZDI5NzQ1YWZlZDA2YmEXMjMNDWYmQ1Y2E5MDJkMDIxIiwidWlkIjo1NGU4ZDkyZDI0MTUwZC00NGV1LTkxZWQtZDg5MmEwNTMwYTYgIn0.Uk6DFifPZxMgjrXprfKiFsgRKCjJr1ximGJw_oxLvr4
Sec-ch-ua-platform: "Windows"
Accept: */*
Origin: https://curiositystream.com
Sec-Fetch-Site: same-site
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: https://curiositystream.com/
Accept-Encoding: gzip, deflate, br

```

For example, as shown, the CuriosityStream platform takes into account the browser type, operating system, and language in providing content to a user. Upon information and belief, the CuriosityStream platform, through at least its use of the Akamai CDN, which upon information and belief is a licensed and controlled component of the CuriosityStream platform, along with the CuriosityStream client applications configures video content that is requested by the user to include code specific to a browser of the user computer identified based upon one or more attributes of the user computer.

107. As in claim 1 of the '004 Patent, the Accused Product causes communication of the content package (e.g., video content) to the network device (e.g., a computer). For example, Curiosity streams video content to the browser of a computer.



<https://curiositystream.com/video/6218> (last visited 10/07/2022).

Headers | TextView | SyntaxView | WebForms | HexView | Auth | Cookies | Raw | JSON | XML

GET https://cdn-s3-akm.curiositystream.com/bitmovin-outputs/WildLogic_HotTuna_ENG_f2997_HD_4227.mp4/fe0a1059ff5885bf85098785bc9f6365.mpd HTTP/1.1
Host: cdn-s3-akm.curiositystream.com

Connection: keep-alive
sec-ch-ua: "Chromium";v="104", " Not A;Brand";v="99", "Google Chrome";v="104"
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (windows NT 10.0; win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/104.0.0.0 Safari/537.36
sec-ch-ua-platform: "Windows"
Accept: */*
Origin: <https://curiositystream.com>
Sec-Fetch-Site: same-site
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: <https://curiositystream.com/>
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9,zh-TW;q=0.8,zh;q=0.7

Find... (press Ctrl+Enter to highlight all) View in Notes

Transformer | Headers | TextView | SyntaxView | ImageView | HexView | WebView | Auth | Caching | Cookies

Raw | JSON | XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<MPD id="7b064dfe-67b5-43b8-ae6d-e81a5ec0d613" type="static"
xmlns="urn:mpeg:dash:schema:mpd:2011" profiles="urn:mpeg:dash:profile:isoff-
live:2011,urn:com:dashif:dash264" xmlns:ns2="http://www.w3.org/1999/xlink"
minBufferTime="POYOMODTOHOM2.000S" mediaPresentationDuration="POYOMODTOH52M6.724S"><Period
id="5cbe54b6-c0fb-48bc-bce0-2b1c6a264685"><AdaptationSet segmentAlignment="true"
mimeType="video/mp4"><Representation id="14004476-fadf-4ecb-ba7d-dffdcdc3ef20" width="640"
codecs="avc1.640016" height="360" bandwidth="145000" frameRate="30000/1001"><SegmentTemplate
media="video/h264/360/145000/base/seg $Number$.m4s" duration="120120" timescale="30000"
startNumber="0" initialization="video/h264/360/145000/base/init.mp4"/></Representation>
<Representation id="1757c519-a8dd-4d3e-885b-3a6858d76f13" width="640" codecs="avc1.640016"
height="360" bandwidth="275500" frameRate="30000/1001"><SegmentTemplate
media="video/h264/360/275500/base/seg $Number$.m4s" duration="120120" timescale="30000"
startNumber="0" initialization="video/h264/360/275500/base/init.mp4"/></Representation>
<Representation id="bc898879-710c-4149-971f-dcc50faa7f51" width="854" codecs="avc1.640016"
height="480" bandwidth="523450" frameRate="30000/1001"><SegmentTemplate
media="video/h264/480/523450/base/seg $Number$.m4s" duration="120120" timescale="30000"
startNumber="0" initialization="video/h264/480/523450/base/init.mp4"/></Representation>
<Representation id="320367b1-72e6-4b51-8259-d161f86b6be7" width="960" codecs="avc1.640016"
```



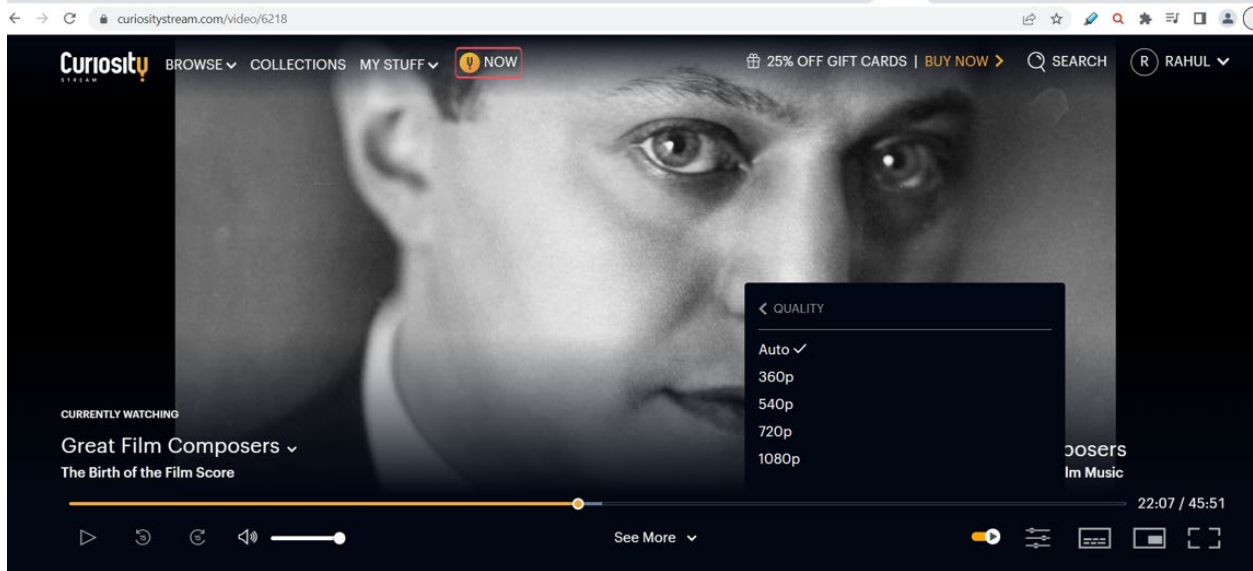
```

height="540" bandwidth="821348" frameRate="30000/1001"><SegmentTemplate
media="video/h264/540/821348/base/seq_$Number$.m4s" duration="120120" timescale="30000"
startNumber="0" initialization="video/h264/540/821348/base/init.mp4"/></Representation>
<Representation id="9496d9f0-6022-4606-8f7e-3d0e27c840a2" width="1280" codecs="avc1.64001F"
height="720" bandwidth="1304659" frameRate="30000/1001"><SegmentTemplate
media="video/h264/720/1304659/base/seq_$Number$.m4s" duration="120120" timescale="30000"
startNumber="0" initialization="video/h264/720/1304659/base/init.mp4"/></Representation>
<Representation id="9193ba9f-b258-4446-b848-dec220f99fd4" width="1280" codecs="avc1.64001F"
height="720" bandwidth="2478852" frameRate="30000/1001"><SegmentTemplate
media="video/h264/720/2478852/base/seq_$Number$.m4s" duration="120120" timescale="30000"
startNumber="0" initialization="video/h264/720/2478852/base/init.mp4"/></Representation>
<Representation id="711cfb60-ff22-4cdb-9d70-b9cf8c16057b" width="1920" codecs="avc1.640028"
height="1080" bandwidth="3922908" frameRate="30000/1001"><SegmentTemplate
media="video/h264/1080/3922908/base/seq_$Number$.m4s" duration="120120" timescale="30000"
startNumber="0" initialization="video/h264/1080/3922908/base/init.mp4"/></Representation>
<Representation id="37ae9eb0-e6fc-475e-9a5f-0901c726bfe2" width="1920" codecs="avc1.640028"
height="1080" bandwidth="6208201" frameRate="30000/1001"><SegmentTemplate
media="video/h264/1080/6208201/base/seq_$Number$.m4s" duration="120120" timescale="30000"
startNumber="0" initialization="video/h264/1080/6208201/base/init.mp4"/></Representation>
</AdaptationSet><AdaptationSet lang="en" label="English" segmentAlignment="true"
mimeType="audio/mp4"><Role schemeIdUri="urn:mpeg:dash:role:2011" value="main"/>
<Representation id="11d7c54c-5f28-4645-b3fa-76463776dced" codecs="mp4a.40.2"
bandwidth="160000" audioSamplingRate="48000"><SegmentTemplate media="audio/full/en/base/seq_
$Number$.m4s" duration="192000" timescale="48000" startNumber="0"
initialization="audio/full/en/base/init.mp4"/></AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
</Representation></AdaptationSet><AdaptationSet mimeType="text/vtt" lang="en"
label="English"><Role schemeIdUri="urn:mpeg:dash:role:2011" value="main"/><Representation
id="caption_en" bandwidth="256"><BaseURL>subtitles/en/full-1617717983312.vtt</BaseURL>
</Representation></AdaptationSet></Period>
</MPD>

```

For example, as show, CuriosityStream streams video content to a browser. Upon information and belief, the CuriosityStream platform, through at least its use of the Akamai CDN, which upon information and belief is a licensed and controlled component of the CuriosityStream platform, along with the CuriosityStream client applications sends the video content to the browser of a user's computer.

108. As in claim 1 of the '004 Patent, in the Accused Product, the one or more attributes include one or more of basic operating characteristics of the at least one network device, a language attribute, a bandwidth attribute, a firewall attribute, or a permissions attribute. For example, the Accused Product has an "auto" setting for determining bandwidth.



<https://curiositystream.com/video/6218> (last visited 10/07/2022).

Headers | TextView | SyntaxView | WebForms | HexView | Auth | Cookies | Raw | JSON | XML

```
GET https://cdn-s3-akm.curiositystream.com/bitmovin-outputs/WildLogic_HotTuna_ENG_f2997_HD_4227.mp4/fe0a1059ff5885bf85098785bc9f6365.mpd HTTP/1.1
Host: cdn-s3-akm.curiositystream.com
Connection: keep-alive
sec-ch-ua: "Chromium";v="104", " Not A;Brand";v="99", "Google Chrome";v="104"
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (windows NT 10.0; win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/104.0.0.0 Safari/537.36
sec-ch-ua-platform: "Windows"
Accept: */*
Origin: https://curiositystream.com
Sec-Fetch-Site: same-site
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: https://curiositystream.com/
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9,zh-TW;q=0.8,zh;q=0.7
```

Find... (press Ctrl+Enter to highlight all) View in Notes

Transformer | Headers | TextView | SyntaxView | ImageView | HexView | WebView | Auth | Caching | Cookies

Raw | JSON | XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<MPD id="7b064dfe-67b5-43b8-ae6d-e81a5ec0d613" type="static"
  xmlns="urn:mpeg:dash:schema:mpd:2011" profiles="urn:mpeg:dash:profile:isoff-live:2011,urn:com:dashif:dash264" xmlns:xs="http://www.w3.org/1999/xlink"
  minBufferTime="POYOMODTOHOM2.000S" mediaPresentationDuration="POYOMODTOH52M6.724S"><Period
  id="5cbe54b6-c0fb-48bc-bce0-2b1c6a264685"><AdaptationSet segmentAlignment="true"
  mimeType="video/mp4"><Representation id="14004476-fadf-4ecb-ba7d-dffdcdc3ef20" width="640"
  codecs="avc1.640016" height="360" bandwidth="145000" frameRate="30000/1001"><SegmentTemplate
  media="video/h264/360/145000/base/seg_${Number$.m4s}" duration="120120" timescale="30000"
  startNumber="0" initialization="video/h264/360/145000/base/init.mp4"/></Representation>
  <Representation id="1757c519-a8dd-4d3e-885b-3a6858d76f13" width="640" codecs="avc1.640016"
  height="360" bandwidth="275500" frameRate="30000/1001"><SegmentTemplate
  media="video/h264/360/275500/base/seg_${Number$.m4s}" duration="120120" timescale="30000"
  startNumber="0" initialization="video/h264/360/275500/base/init.mp4"/></Representation>
  <Representation id="bc898879-710c-4149-971f-dcc50faa7f51" width="854" codecs="avc1.640016"
  height="480" bandwidth="523450" frameRate="30000/1001"><SegmentTemplate
  media="video/h264/480/523450/base/seg_${Number$.m4s}" duration="120120" timescale="30000"
  startNumber="0" initialization="video/h264/480/523450/base/init.mp4"/></Representation>
  <Representation id="320367b1-72e6-4b51-8259-d161f86b6be7" width="960" codecs="avc1.640016"
```

Additionally, for example, the CuriosityStream platform takes into account operating system, language, and bandwidth in providing video content to a user. Upon information and belief, the CuriosityStream platform, through at least its use of the Akamai CDN, which upon information and belief is a licensed and controlled component of the CuriosityStream platform, along with the CuriosityStream client applications sends the video content based on the one or more attributes including at least operating system, language, and bandwidth.

109. Defendant makes, uses, sells, and/or offers to sell the Accused Products which practices at least claim 1 of the '004 Patent.

110. In violation of 35 U.S.C. § 271, Defendant has been directly infringing the '004 Patent, including through its own use, testing, and sell of the Accused Products.

111. Defendant has had knowledge of infringement of the '004 Patent at least as of the service of the present Complaint.

112. Defendant has directly infringed at least one claim of the '004 Patent by making, using, offering for sale, and selling the Accused Product without authority in the United States. As a direct and proximate result of Defendant's direct infringement of the '004 Patent, DMS has been damaged.

113. By engaging in the conduct described herein, Defendant has injured DMS and is thus liable for infringement of the '004 Patent, pursuant to 35 U.S.C. § 271.

114. Defendant has committed these acts of infringement without license or authorization.

115. As a result of Defendant's infringement of the '004 Patent, Plaintiff has suffered monetary damages and is entitled to a monetary judgment in an amount adequate to compensate for Defendant's past infringement, together with interests and costs.

116. Plaintiff reserves the right to modify its infringement theories as discovery progresses in this case; it shall not be estopped for infringement contention or claim construction purposes by the claim chart that it provides with this Complaint.

COUNT V: INFRINGEMENT OF U.S. PATENT NO. 8,046,672

117. Plaintiff incorporates by reference and re-alleges the foregoing paragraphs of this Complaint as if fully set forth herein.

118. Defendant has directly infringed at least claim 1 of the '672 Patent in violation of 35 U.S.C. § 271 *et seq.*, by making, using, offering for sale, or selling in the United States, and/or importing into the United States without authority or license the Accused Product. Further, upon information and belief, Defendant directly infringes at least claim 1 of the '672 Patent through hardware and software systems, such as content delivery systems, that it owns, controls, or licenses.

119. Defendant has directly infringed at least claim 1 of the '672 Patent by performing all the limitation of that claim. For example, claim 1 of the '672 Patent recites:

A method comprising:

receiving an indication over a network, in response to a request for access to a rich media presentation by an internet browser on a network device, that the internet browser on the network device has requested access to the rich media presentation;


in response to the receiving an indication, detecting one or more attributes of one or both of rich media capabilities associated with the internet browser or rich media capabilities associated with the network device; and

selecting a rich media presentation to be sent to the internet browser from among a plurality of rich media presentations based on the one or more attributes that are detected, wherein a selected rich media presentation includes a media package selected based on the one or more attributes that are detected and a virtual player configured to play the media package on the network device.

120. As in claim 1 of the '672 Patent, the Accused Product receives an indication over a network, in response to a request for access to a rich media presentation (*e.g.*, a video) by an

internet browser on a network device (e.g., a computer), that the internet browser on the network device has requested access to the rich media presentation. Upon information and belief, the Defendant has licensed to use and exercises control over its use of the Akamai Content Delivery Network to implement video streaming services.

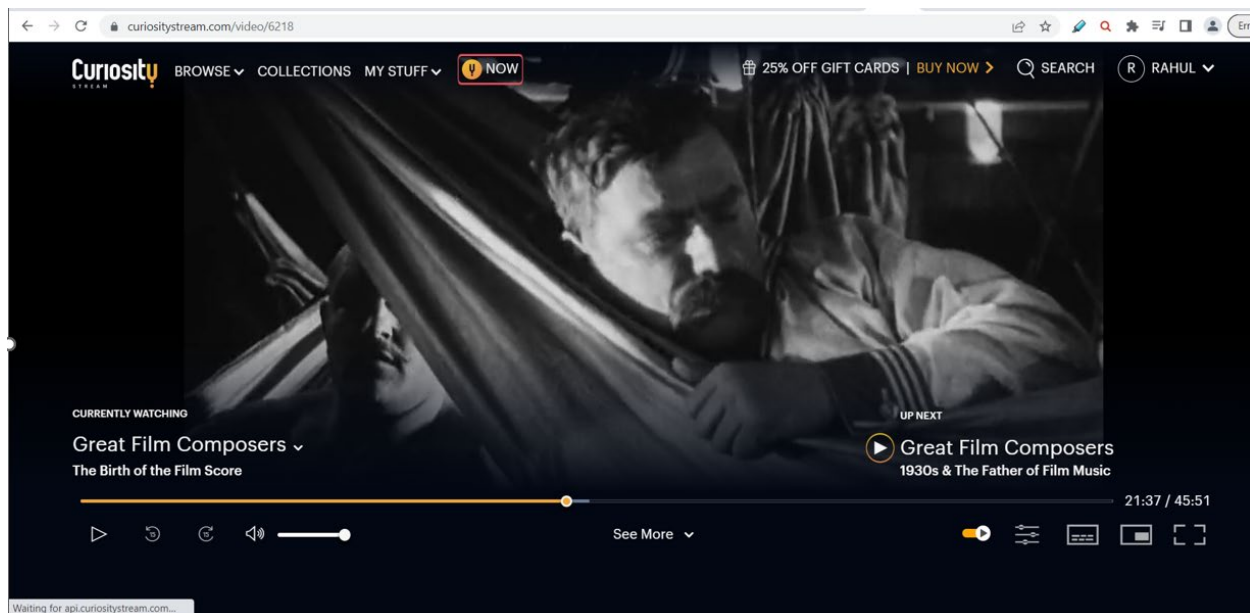
What platforms and devices do you currently support?

 Curiosity Stream Support
2 days ago · Updated

You can enjoy CuriosityStream on the following platforms:

- Any laptop, desktop, PC, or Mac via an up-to-date web browser and operating system
- Amazon Fire TV Stick and Amazon Fire TV version 5.0 and up (2014 or later)
- Roku 2, Roku 3, Roku 4, Roku TV, and Roku Streaming Stick version 8.1 and up
- Google Chromecast via either the Chrome browser, Android or iOS devices
- Chrome extension for easy access to the web app
- Android device running version 5.0 and up
- Android TV running Android version 5.0 and up
- Apple iPhones, iPods, and iPads iOS 11 and up
- Apple TV 3rd generation and earlier via Air Play
- Apple TV 4th generation HD and 4K Apple TV
- Xbox One
- PlayStation 4
- Available Smart TVs Market apps (recommended devices from 2016 and later): LG, Samsung, Sony/Android TV, TiVo and VIZIO

<https://help.curiositystream.com/hc/en-us/articles/209260738-What-platforms-and-devices-do-you-currently-support-> (last visited 10/06/2022).



<https://curiositystream.com/video/6218> (last visited 10/11/2022).

Figure 1 shows a possible deployment architecture in which the formats defined in this document may be used. Boxes with solid lines indicate devices that are mentioned in this document as they host or process the formats defined in this document whereas dashed boxes are conceptual or transparent. This document deals with the definition of formats that are accessible on the interface to the DASH Client, indicated by the solid lines. Any other formats or interfaces are outside the scope of this document. In the considered deployment scenario, it is assumed that the DASH Client has access to an MPD. The MPD provides sufficient information for the DASH Client to provide a streaming service to the user by requesting Segments from an HTTP server and demultiplexing, decoding and rendering the included media streams.

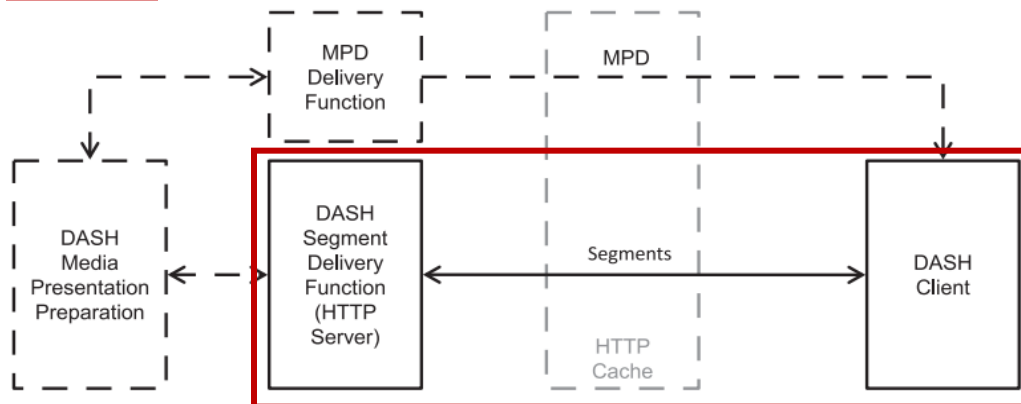
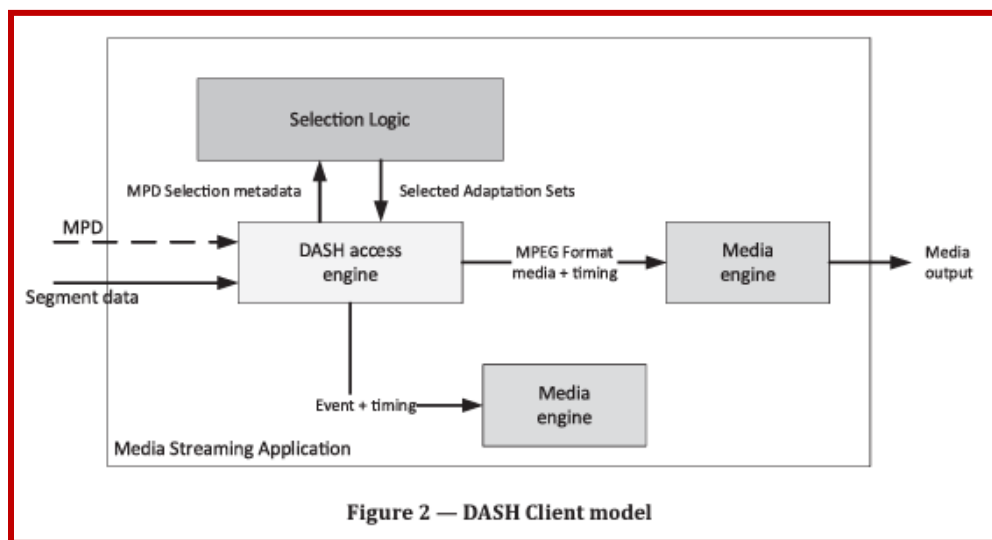


Figure 1 — Example system for DASH formats

Although the formats are initially designed to be used in the above deployment scenario, their application is obviously not restricted to this scenario. The particular aspect on "HTTP" in DASH is the usage of HTTP-URLs in the MPD for the purpose to refer to Segments. The usage of HTTP-URLs enables unique location information and it provides well-defined methods to access the resources, in particular HTTP GET and HTTP partial GET.

4.2 DASH Client model

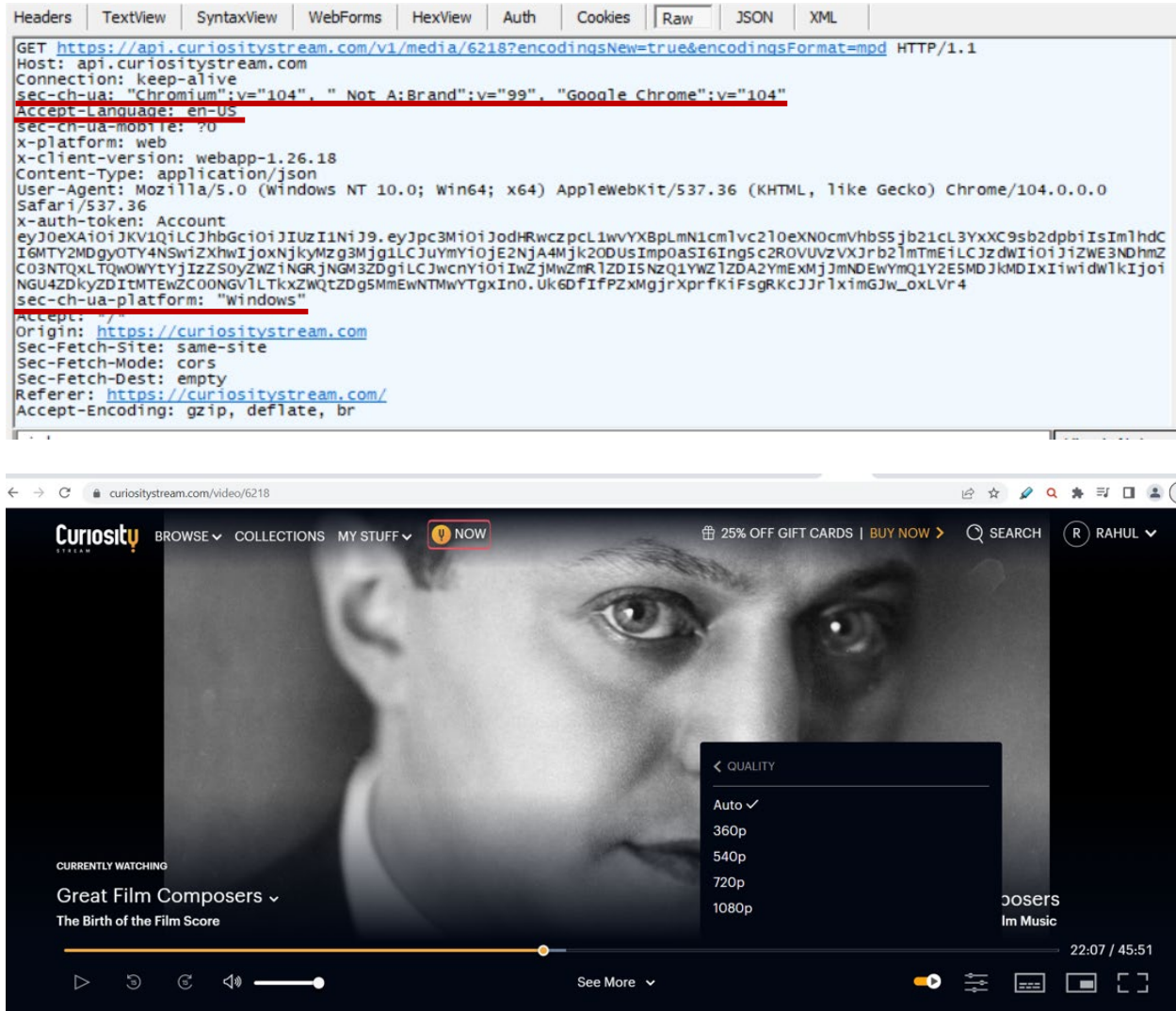
The design of the formats defined in this document is based on the informative client model as shown in [Figure 2](#). The figure illustrates the logical components of a conceptual DASH Client model and the relation to other components in a media streaming application. In this figure, the DASH access engine receives the Media Presentation Description (MPD), constructs and issues requests and receives Segments or parts of Segments. The DASH Client may use metadata provided in the MPD for the selection of media components by communication with the media streaming application. Such metadata may for example include codec capability information, language codes, accessibility information and other information for the selection of media components. In the context of this document, the output of the DASH access engine consists of media in MPEG container formats (ISO/IEC 14496-12 ISO base media file format or ISO/IEC 13818-1 MPEG-2 Transport Stream), or parts thereof, together with timing information that maps the internal timing of the continuous media to the timeline of the Media Presentation. In [Annex F](#), guidance on enabling the use of this document with other container formats is provided. In addition, the DASH access client may also receive and extract Events that are related to the media time. The events may be processed in the DASH Client or may be forwarded to an event processing application in the execution environment of the DASH Client.



ISO/IEC 23009-1:2019.

For example, as shown, the Curiosity Stream platform allows its users to use an internet browser to select and stream video content. Upon information and belief, the CuriosityStream platform, through at least its use of the Akamai CDN, which upon information and belief is a licensed and controlled component of the CuriosityStream platform, along with the CuriosityStream client applications receives a request for access to a video content from an internet browser on a user's computer, that the internet browser on the user's computer has requested access to the video content.

121. As in claim 1 of the '672 Patent, in response to the receiving an indication, the Accused Product, on information and belief, detects one or more attributes of one or both of rich media capabilities associated with the internet browser or rich media capabilities associated with the network device. For example, the Accused Product detects attributes of the computer, *e.g.*, operating system platform and bandwidth.



<https://curiositystream.com/video/6218> (last visited 10/11/2022).

The screenshot shows a web browser's developer tools with the 'Headers' tab selected. The top section displays the request headers for a GET request to a video file on a CDN. The 'sec-ch-ua' and 'sec-ch-ua-mobile' headers are highlighted in red. Below the headers, the 'Find...' search bar is visible. The bottom section shows the raw XML response, with several attributes like 'bandwidth' and 'height' highlighted in red.

```

GET https://cdn-s3-akm.curiositystream.com/bitmovin-outputs/WildLogic_HotTuna_ENG_f2997_HD_4227.mp4/fe0a1059ff5885bf85098785bc9f6365.mpd HTTP/1.1
Host: cdn-s3-akm.curiositystream.com
Connection: keep-alive
sec-ch-ua: "Chromium";v="104", " Not A:Brand";v="99", "Google Chrome";v="104"
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/104.0.0.0 Safari/537.36
sec-ch-ua-platform: "Windows"
Accept: */*
Origin: https://curiositystream.com
Sec-Fetch-Site: same-site
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: https://curiositystream.com/
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9,zh-TW;q=0.8,zh;q=0.7

Find... (press Ctrl+Enter to highlight all) View in Notes

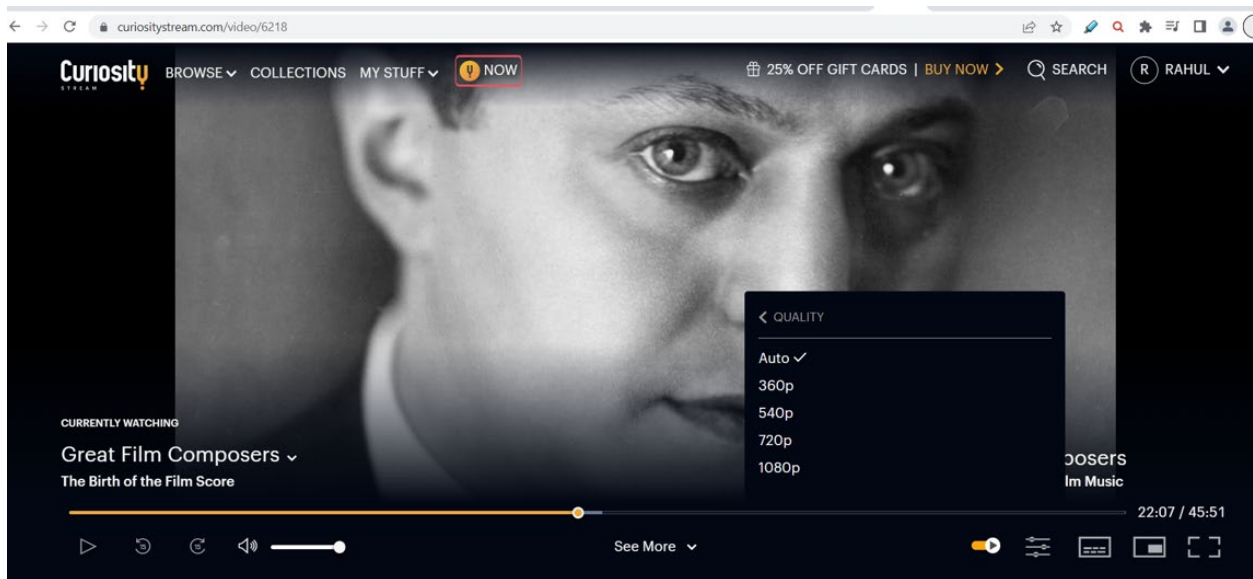
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<MPD id="7b064dfe-67b5-43b8-ae6d-e81a5ec0d613" type="static"
xmlns="urn:mpeg:dash:schema:mpd:2011" profiles="urn:mpeg:dash:profile:isoff-
live:2011,urn:com:dashif:dash264" xmlns:ns2="http://www.w3.org/1999/xlink"
minBufferTime="POYOMODT0H0M2.000S" mediaPresentationDuration="POYOMODT0H52M6.724S"><Period
id="5cbe54b6-c0fb-48bc-bce0-2b1c6a264685"><AdaptationSet segmentAlignment="true"
mimeType="video/mp4"><Representation id="14004476-fadf-4ecb-ba7d-dffdcdc3ef20" width="640"
codecs="avc1.640016" height="360" bandwidth="145000" frameRate="30000/1001"><SegmentTemplate
media="video/h264/360/145000/base/seg_$Number$.m4s" duration="120120" timescale="30000"
startNumber="0" initialization="video/h264/360/145000/base/init.mp4"/></Representation>
<Representation id="1757c519-a8dd-4d3e-885b-3a6858d76f13" width="640" codecs="avc1.640016"
height="360" bandwidth="275500" frameRate="30000/1001"><SegmentTemplate
media="video/h264/360/275500/base/seg_$Number$.m4s" duration="120120" timescale="30000"
startNumber="0" initialization="video/h264/360/275500/base/init.mp4"/></Representation>
<Representation id="bc898879-710c-4149-971f-dcc50faa7f51" width="854" codecs="avc1.640016"
height="480" bandwidth="523450" frameRate="30000/1001"><SegmentTemplate
media="video/h264/480/523450/base/seg_$Number$.m4s" duration="120120" timescale="30000"
startNumber="0" initialization="video/h264/480/523450/base/init.mp4"/></Representation>
<Representation id="320367b1-72e6-4b51-8259-d161f86b6be7" width="960" codecs="avc1.64001F"

```

For example, the CuriosityStream platform takes into account operating system, language, and bandwidth in providing video content to a user. Upon information and belief, the CuriosityStream platform, through at least its use of the Akamai CDN, which upon information and belief is a licensed and controlled component of the CuriosityStream platform, along with the CuriosityStream client applications, in response to the receiving a request for video content, detects one or more attributes of video capabilities associated with the internet browser or the user's computer, such as language, bandwidth, operating system, and browser type.

122. As in claim 1 of the '672 Patent, the Accused Product, on information and belief, selects a rich media presentation to be sent to the internet browser from among a plurality of rich media presentations based on the one or more attributes that are detected, wherein a selected rich

media presentation includes a media package selected based on the one or more attributes that are detected and a virtual player configured to play the media package on the network device. For example, the Accused Product sends video to the browser that is selected based on detected attributes.



<https://curiositystream.com/video/6218> (last visited 10/11/2022).

Headers	TextView	SyntaxView	WebForms	HexView	Auth	Cookies	Raw	JSON	XML
<pre>GET https://cdn-s3-akm.curiositystream.com/bitmovin-outputs/WildLogic_HotTuna_ENG_f2997_HD_4227.mp4/fe0a1059ff5885bf85098785bc9f6365.mpd HTTP/1.1 Host: cdn-s3-akm.curiositystream.com Connection: keep-alive sec-ch-ua: "Chromium";v="104", " Not A;Brand";v="99", "Google Chrome";v="104" sec-ch-ua-mobile: ?0 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/104.0.0.0 Safari/537.36 sec-ch-ua-platform: "Windows" Accept: */* Origin: https://curiositystream.com Sec-Fetch-Site: same-site Sec-Fetch-Mode: cors Sec-Fetch-Dest: empty Referer: https://curiositystream.com/ Accept-Encoding: gzip, deflate, br Accept-Language: en-US,en;q=0.9,zh-TW;q=0.8,zh;q=0.7</pre>									
Find... (press Ctrl+Enter to highlight all)								View in Notes	
Transformer	Headers	TextView	SyntaxView	ImageView	HexView	WebView	Auth	Caching	Cookies
Raw	JSON	XML							
<pre><?xml version="1.0" encoding="UTF-8" standalone="yes"?> <MPD id="7b064dfe-67b5-43b8-ae6d-e81a5ec0d613" type="static" xmlns="urn:mpeg:dash:schema:mpd:2011" profiles="urn:mpeg:dash:profile:isoff- live:2011,urn:com:dashif:dash264" xmlns:ns2="http://www.w3.org/1999/xhtml" minBufferTime="POYOMODTOM2.000S" mediaPresentationDuration="POYOMODTOM2M6.724S"><Period id="5cbe54b6-c0fb-48bc-bce0-2b1c6a264685"><AdaptationSet segmentAlignment="true" mimeType="video/mp4"><Representation id="14004476-fadf-4ecb-ba7d-dffdcdc3ef20" width="640" codecs="avc1.640016" height="360" bandwidth="145000" frameRate="30000/1001"><SegmentTemplate media="video/h264/360/145000/base/seg_\$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/360/145000/base/init.mp4"/></Representation> <Representation id="1757c519-a8dd-4d3e-885b-3a6858d76f13" width="640" codecs="avc1.640016" height="360" bandwidth="275500" frameRate="30000/1001"><SegmentTemplate media="video/h264/360/275500/base/seg_\$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/360/275500/base/init.mp4"/></Representation> <Representation id="bc898879-710c-4149-971f-dcc50faa7f51" width="854" codecs="avc1.640016" height="480" bandwidth="523450" frameRate="30000/1001"><SegmentTemplate media="video/h264/480/523450/base/seg_\$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/480/523450/base/init.mp4"/></Representation> <Representation id="320367b1-72e6-4b51-8259-d161f86b6be7" width="960" codecs="avc1.64001F" height="540" bandwidth="821348" frameRate="30000/1001"><SegmentTemplate media="video/h264/540/821348/base/seg_\$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/540/821348/base/init.mp4"/></Representation> <Representation id="9496d9f0-6022-4606-8f7e-3d0e27c840a2" width="1280" codecs="avc1.64001F" height="720" bandwidth="1304659" frameRate="30000/1001"><SegmentTemplate media="video/h264/720/1304659/base/seg_\$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/720/1304659/base/init.mp4"/></Representation> <Representation id="9193ba9f-b258-4446-b848-dec220f99fd4" width="1280" codecs="avc1.64001F" height="720" bandwidth="2478852" frameRate="30000/1001"><SegmentTemplate media="video/h264/720/2478852/base/seg_\$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/720/2478852/base/init.mp4"/></Representation> <Representation id="711cfb60-ff22-4cdb-9d70-b9cf8c16057b" width="1920" codecs="avc1.640028" height="1080" bandwidth="3922908" frameRate="30000/1001"><SegmentTemplate media="video/h264/1080/3922908/base/seg_\$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/1080/3922908/base/init.mp4"/></Representation> <Representation id="37ae9eb0-e6fc-475e-9a5f-0901c726bfe2" width="1920" codecs="avc1.640028" height="1080" bandwidth="6208201" frameRate="30000/1001"><SegmentTemplate media="video/h264/1080/6208201/base/seg_\$Number\$.m4s" duration="120120" timescale="30000" startNumber="0" initialization="video/h264/1080/6208201/base/init.mp4"/></Representation> </AdaptationSet><AdaptationSet lang="en" label="English" segmentAlignment="true" mimeType="audio/mp4"><Role schemeIdUri="urn:mpeg:dash:role:2011" value="main"/> <Representation id="11d7c54c-5f28-4645-b3fa-76463776dced" codecs="mp4a.40.2" bandwidth="160000" audioSamplingRate="48000"><SegmentTemplate media="audio/full/en/base/seg_ \$Number\$.m4s" duration="192000" timescale="48000" startNumber="0" initialization="audio/full/en/base/init.mp4"/></Representation> </AdaptationSet><AdaptationSet mimeType="text/vtt" lang="en" label="English"><Role schemeIdUri="urn:mpeg:dash:role:2011" value="main"/><Representation id="caption_en" bandwidth="256"><BaseURL>subtitles/en/full-1617717983312.vtt</BaseURL> </Representation></AdaptationSet></Period> </MPD></pre>									

For example, the CuriosityStream platform takes into account operating system, language, and bandwidth in providing video content to a user. Upon information and belief, the CuriosityStream platform, through at least its use of the Akamai CDN, which upon information and belief is a licensed and controlled component of the CuriosityStream platform, along with the CuriosityStream client applications selects video content to be sent to the internet browser based on the one or more attributes, such as language, browser, bandwidth, or operating system, that are detected and provides video content that is capable of being rendered in the CuriosityStream application being run on the browser.

123. Defendant makes, uses, sells, and/or offers to sell the Accused Products which practices at least claim 1 of the '672 Patent.

124. In violation of 35 U.S.C. § 271, Defendant has been directly infringing the '672 Patent, including through its own use, testing, and sell of the Accused Products.

125. Defendant has had knowledge of infringement of the '672 Patent at least as of the service of the present Complaint.

126. Defendant has directly infringed and continues to infringe at least one claim of the '672 Patent by making, using, offering for sale, and selling the Accused Product without authority in the United States. As a direct and proximate result of Defendant's direct infringement of the '672 Patent, DMS has been damaged.

127. By engaging in the conduct described herein, Defendant has injured DMS and is thus liable for infringement of the '672 Patent, pursuant to 35 U.S.C. § 271.

128. Defendant has committed these acts of infringement without license or authorization.

129. As a result of Defendant's infringement of the '672 Patent, Plaintiff has suffered monetary damages and is entitled to a monetary judgment in an amount adequate to compensate for Defendant's past infringement, together with interests and costs.

130. Plaintiff reserves the right to modify its infringement theories as discovery progresses in this case; it shall not be estopped for infringement contention or claim construction purposes by the claim chart that it provides with this Complaint.

DEMAND FOR JURY TRIAL

131. DMS demands a trial by jury of any and all causes of action.

PRAYER FOR RELIEF

WHEREFORE, DMS respectfully requests:

- a. That Judgment be entered that Defendant has infringed one or more claims of the '811 Patent;
- b. That Judgment be entered that Defendant has infringed one or more claims of the '922 Patent;
- c. That Judgment be entered that Defendant has infringed one or more claims of the '714 Patent;
- d. That Judgment be entered that Defendant has infringed one or more claims of the '672 Patent;
- e. That Judgment be entered that Defendant has infringed one or more claims of the '004 Patent;
- f. An award of damages pursuant to 35 U.S.C. §284, sufficient to compensate Plaintiff for the Defendant's past infringement and any continuing or future infringement;

- g. An assessment of pre-judgment and post-judgment interest and costs against Defendant, together with an award of such interest and costs, in accordance with 35 U.S.C. §284;
- h. That Defendant be directed to pay enhanced damages, including Plaintiff's attorneys' fees incurred in connection with this lawsuit pursuant to 35 U.S.C. §285; and
- i. That Plaintiff be granted such other and further relief as this Court may deem just and proper.

Dated: January 12, 2023

Respectfully submitted,

/s/ Stamatios Stamoulis

Stamatios Stamoulis (#4606)
Richard C. Weinblatt (#5080)
STAMOULIS & WEINBLATT LLC
800 N. West Street, Third Floor
Wilmington, DE 19801
Telephone: (302) 999-1540
Facsimile: (302) 762-1688
stamoulis@swdelaw.com
weinblatt@swdelaw.com

Mark S. Raskin
Michael S. DeVincenzo
Daniel C. Miller
KING & WOOD MALLESONS LLP
500 5th Avenue, 50th Floor
New York, New York 10110
(212) 319-4755
mark.raskin@us.kwm.com
michael.devincenzo@us.kwm.com

Attorneys for Plaintiff
Distributed Media Solutions, LLC